Ортогональные вейвлеты Добеши

Файл содержит теорию построения банков фильтров ортогонального вейвлет-преобразования, средства расчета фильтров и их применение для вейвлет преобразования.

Основные обозначения

Данная блок схема представляет собой 2-х канальный банк фильтр вейвлет преобразования

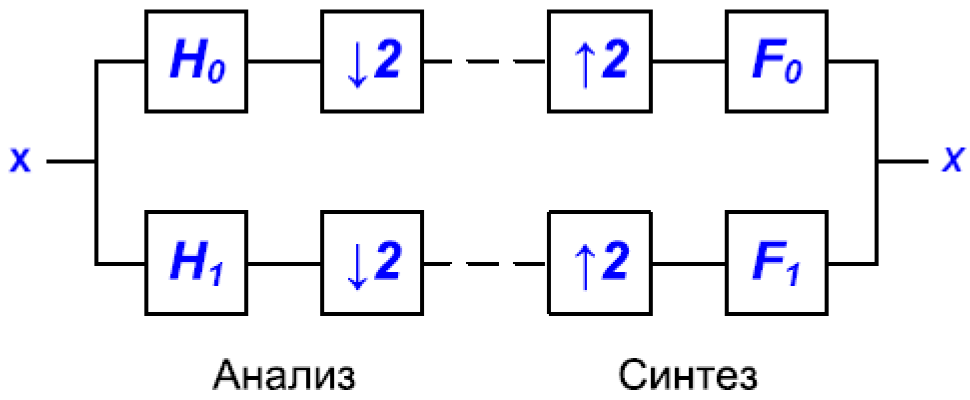

Рис.1 Двухканальный банк фильтр вейвлет преобразования.

Банк фильтров анализа разделяет входной сигнал **x** на два с помощью фильтров анализа с импульсными характеристиками H_0 и H_1. После фильтров каждый 2-ой отсчет сигнала отбрасывается.

В обоих каналах блока синтеза после каждого входного отсчета вставляется нулевой отсчет, далее эти сигналы поступают на фильтры синтеза F_0 и F_1 и суммируются.

Стоит задача найти такие фильтры, чтобы 2-х канальный банк фильтр на выходе восстанавливал сигнал **x** абсолютно точно.

Точное восстановление имеет место при выполнении следующих двух равенств:

$$F(z)_0 \cdot H(z)_0 + F(z)_1 \cdot H(z)_1 = 2z^{-N} \qquad (1)$$

$$F(z)_0 \cdot H(-z)_0 + F(z)_1 \cdot H(-z)_1 = 0 \qquad (2)$$

$H(z)_0$, $H(z)_1$ и $F(z)_0$, $F(z)_1$ - это z-преобразование импульсной характеристики фильтров анализа и синтеза.

Для выполнения этих 2-х равенств определим фильтры анализа и синтеза следующим образом:

Z-преобразование импульсной характеристики фильтра:	***Импульсная характеристика фильтра:***
$H(z)_0 = \sum_{k=0}^{N} h(k)\cdot z^{-k}$	$h(0), h(1), \ldots, h(N-1), h(N)$
$H(z)_1 = -z^{-N}\cdot H\left(-z^{-1}\right)_0 = \sum_{k=0}^{N} (-1)^k\cdot h(N-k)\cdot z^{-k}$	$h(N), -h(N-1), \ldots, h(1), -h(0)$
$F(z)_0 = H(-z)_1 = \sum_{k=0}^{N} h(N-k)\cdot z^{-k}$	$h(N), h(N-1), \ldots, h(1), h(0)$
$F(z)_1 = -H(-z)_0 = \sum_{k=0}^{N} (-1)^{k+1}\cdot h(k)\cdot z^{-k}$	$-h(0), h(1), \ldots, -h(N-1), h(N)$

Данный выбор приводит нас к сопряженным квадратурным фильтрам (***Conjugate Quadrature Filters - CQF***). Они также появляются при построении ортогональных вейвлетов Добеши.

Заменой $\mathbf{H(z)_1}$ на $\mathbf{F(-z)_0}$ и $\mathbf{F(z)_1}$ на $\mathbf{-H(-z)_0}$ в формуле (1) получаем:

$$F(z)_0\cdot H(z)_0 - F(-z)_0\cdot H(-z)_0 = 2\cdot z^{-N}$$

Произведя замену $\mathbf{P(z)_0=F(z)_0H(z)_0}$ равенство (1) преобразуется к виду:

$$P(z)_0 - P(-z)_0 = 2\cdot z^{-N}$$

$$P(z)_0 = z^{-N}\cdot H\left(z^{-1}\right)_0\cdot H(z)_0 \qquad P(-z)_0 = -z^{-N}\cdot H\left(-z^{-1}\right)_0\cdot H(-z)_0$$

Умножая $\mathbf{P(z)_0}$ на $\mathbf{z^N}$ получаем следующие соотношения:

$$P(z) = z^N P(z)_0 = H\left(z^{-1}\right)_0\cdot H(z)_0$$

$$P(z) + P(-z) = 2 \qquad (3)$$

Фильтр удовлетворяющий соотношению (3) называется - ***полуполосным фильтром***

Пример

$$P(z) := 1 + \frac{9}{16}\cdot z^{-1} + \frac{9}{16}\cdot z - \frac{1}{16}\cdot z^{-3} - \frac{1}{16}\cdot z^{3}$$

$$P(z) + P(-z) \text{ simplify } \rightarrow 2$$

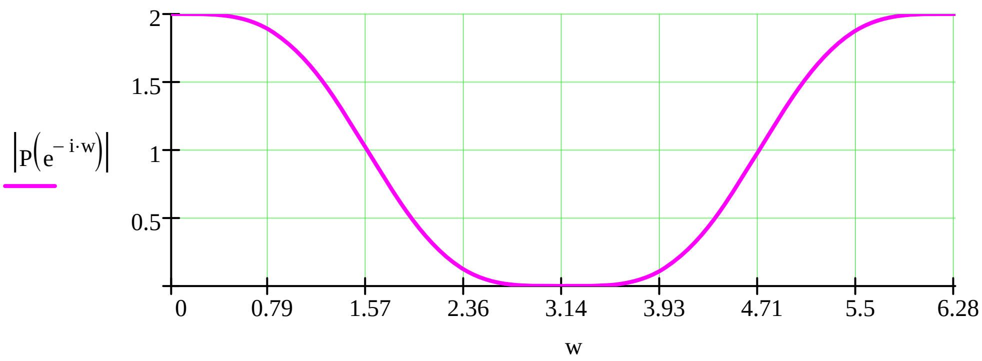

Рис.2 Частотная характеристика полуполосного фильтра.

Заменой **$z=e^{iw}$** - перейдем в частотную область.

Тогда получим, что **$P(e^{iw})=H(e^{-iw})_0 H(e^{iw})_0$** и соответственно **$|P(w)|=|H(w)_0|^2$**.

Таким образом находя **P(z)** и разлагая его на множители мы определим **$H(z)_0$**.

Для ортогональных вейвлетов Добеши с ***p*** нулевыми моментами формула для вычисления **P(z)** имеет вид:

$$P(z) = 2\cdot\left(\frac{1+z}{2}\right)^{P}\cdot\left(\frac{1+z^{-1}}{2}\right)^{P}\cdot\sum_{k=0}^{P-1}\left(\frac{(P+k-1)!}{k!\cdot(P-1)!}\right)\cdot\left(\frac{1-z}{2}\right)^{k}\cdot\left(\frac{1-z^{-1}}{2}\right)^{k}$$

Наличие ***p*** нулевых моментов означает, что **$H(w)_0$** должен иметь нуль порядка ***p*** в точке: **$w=\pi$**. Тогда фильтр **$H(z)_0$** будет обладать **2*p*** - ненулевыми коэффициентами.

$$P(z,p) := 2\left(\frac{1+z}{2}\right)^{p}\cdot\left(\frac{1+z^{-1}}{2}\right)^{p}\cdot\sum_{k=0}^{p-1}\left(\frac{(p+k-1)!}{k!\cdot(p-1)!}\right)\cdot\left(\frac{1-z}{2}\right)^{k}\cdot\left(\frac{1-z^{-1}}{2}\right)^{k}$$

Покажем, что **P(z)** удовлетворяет соотношению (3):

$$p := 10$$

$$P(z,p) + P(-z,p) \text{ simplify } \rightarrow 2$$

Для нахождения $\mathbf{H(z)_0}$ необходимо определить все значения **z** для которых **P(z)=0**.
Всего **P(z)** имеет **4*p*-2** нулей: **2*p*** нулей при **z=-1** и
2*p*-2 нулей, даваемых множителем:

$$B(z,p) := \sum_{k=0}^{p-1} \left(\frac{(p+k-1)!}{k!\cdot(p-1)!}\right)\cdot\left(\frac{1-z}{2}\right)^{k}\cdot\left(\frac{1-z^{-1}}{2}\right)^{k}$$

Таким образом можно вычислить $\mathbf{H(z)_0}$ в 2 шага:

1. Найти все нули **B(z,p)** и выделить из них ***p*-1** нуль, значения которых по модулю меньше **1**.
2. Добавить ***p*** нулей при **z=-1**. Тогда $\mathbf{H(z)_0}$ будет иметь **2*p*-1** нуль.

Расчет коэффициентов вейвлета Добеши

Для вычисления коэффициентов вейвлета Добеши необходимо задать только количество нулевых моментов вейвлета "P"

1) Задаем количество нулевых моментов $p := 7$

2) Находим нули **B(z,p)**

$$B(z,p) := \sum_{k=0}^{p-1} \left(\frac{(p+k-1)!}{k!\cdot(p-1)!}\right)\cdot\left(\frac{1-z}{2}\right)^k\cdot\left(\frac{1-z^{-1}}{2}\right)^k$$

$$SPF := B(z,p) = 0 \text{ solve}, z \rightarrow \begin{pmatrix} .26192301501370458750 - .39318454417261394145\cdot i \\ .26192301501370458750 + .39318454417261394145\cdot i \\ .33406281883084842679 - .20673317774852633826\cdot i \\ .33406281883084842679 + .20673317774852633826\cdot i \\ .35726492400363951991 - 6.5232148056767829483\cdot 10^{-2}\cdot i \\ .35726492400363951991 + 6.5232148056767829483\cdot 10^{-2}\cdot i \\ 1.1735020320374838069 - 1.7615972446260681177\cdot i \\ 1.1735020320374838069 + 1.7615972446260681177\cdot i \\ 2.1645084558154166087 - 1.3394957059284547186\cdot i \\ 2.1645084558154166087 + 1.3394957059284547186\cdot i \\ 2.7087387542989070502 - .49458213106230131661\cdot i \\ 2.7087387542989070502 + .49458213106230131661\cdot i \end{pmatrix}$$

3) Отделяем нули **B(z,p)**, значение по модулю которых меньше **1**

$$ms := 0 .. \mathrm{last}(SPF)$$

$$\mathrm{last}(SPF) = 11$$

$$Vct_{ms} := \mathrm{if}\left(\left|SPF_{ms}\right| < 1, SPF_{ms}, 0\right)$$

4) Формируем полином из нулей, значение которых по модулю меньше **1**

$$Fzp(z) := \frac{\displaystyle\prod_{ms=0}^{2(p-1)-1} \left(z - Vct_{ms}\right)}{z^{p-1}}$$

5) Формируем полином **$H(z)=H(z)_0$** и находим его коффициенты при степенях **z**, так как они определяют значения коэффициентов вейвлета Добеши. Умножение на **z^{1-2p}** необходимо для того, чтобы фильтр **H(z)** был причинным

$$H(z) := Fzp(z)\cdot(z+1)^{p}\cdot z^{1-2p}$$

$$KF := H(z)\ \mathrm{coeffs}, z \ \rightarrow \begin{pmatrix} 4.5434099861808081968\cdot10^{-3} - 1.\cdot10^{-23}\cdot i \\ -2.3141851878181612679\cdot10^{-2} + 2.3\cdot10^{-22}\cdot i \\ 5.517875898974838 90\cdot10^{-3} - 1.1\cdot10^{-22}\cdot i \\ .16121602318154311489 - 4.05\cdot10^{-21}\cdot i \\ -.2128979360334998572 + 1.5\cdot10^{-22}\cdot i \\ -.48848983346652769987 + 5.229\cdot10^{-20}\cdot i \\ 1.0354589881862357 49 + 1.6023\cdot10^{-19}\cdot i \\ .915958096480836279 + 2.4009\cdot10^{-19}\cdot i \\ -2.877717070242550296 + 2.103\cdot10^{-19}\cdot i \\ -1.848454811114268553 8 + 1.10\cdot10^{-19}\cdot i \\ 6.034295342961999298 + 3.2\cdot10^{-20}\cdot i \\ 9.365611471857559014 + 4.\cdot10^{-21}\cdot i \\ 5.09349848430362 \\ 1 \end{pmatrix}$$

6) Нормируем значения коэффициентов и выделяем их действительные значения в обратном порядке для того, чтобы коэффициенты фильтра соответствовали возрастанию отрицательной степени **z**

$$H_0 := \mathrm{reverse}\left(\mathrm{Re}\left(\frac{KF}{|KF|}\right)\right)$$

7) Отображаем ***2p*** - коэффициентов вейвлета Добеши

$H_0 =$

	0
0	0.07785205
1	0.39653932
2	0.72913209
3	0.46978229
4	-0.143906
5	-0.22403618
6	0.07130922
7	0.08061261
8	-0.03802994
9	-0.01657454
10	0.012551
11	0.00042958
12	-0.00180164
13	0.00035371

8) Построение графика импульсной характеристики вейвлета Добеши

$$n := 0 .. \text{last}(H_0)$$

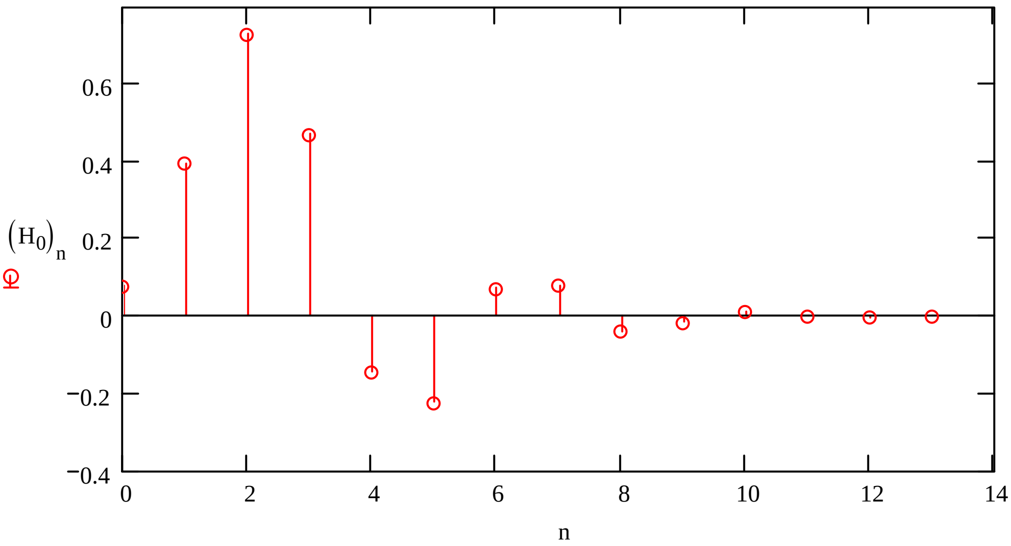

Рис.3 Импульсная характеристика вейвлета Добеши.

9) Построение частотных характеристик верхнего и нижнего канала банк фильтра вейвлет преобразования

$$N := \text{last}(H_0)$$

$$F(w) := \sum_{k=0}^{N} (H_0)_k \cdot e^{-i \cdot w \cdot k}$$

Частотная передаточная функция фильтра с импульсным откликом $\mathbf{H_0}$

$$R(w) := \sum_{k=0}^{N} (-1)^k \cdot (H_0)_{N-k} \cdot e^{-i \cdot w \cdot k}$$

Частотная передаточная функция фильтра с импульсным откликом $\mathbf{H_1}$

$$V(w) := (|F(w)|)^2$$ <-- Частотная характеристика верхнего канала

$$G(w) := (|R(w)|)^2$$ <-- Частотная характеристика нижнего канала

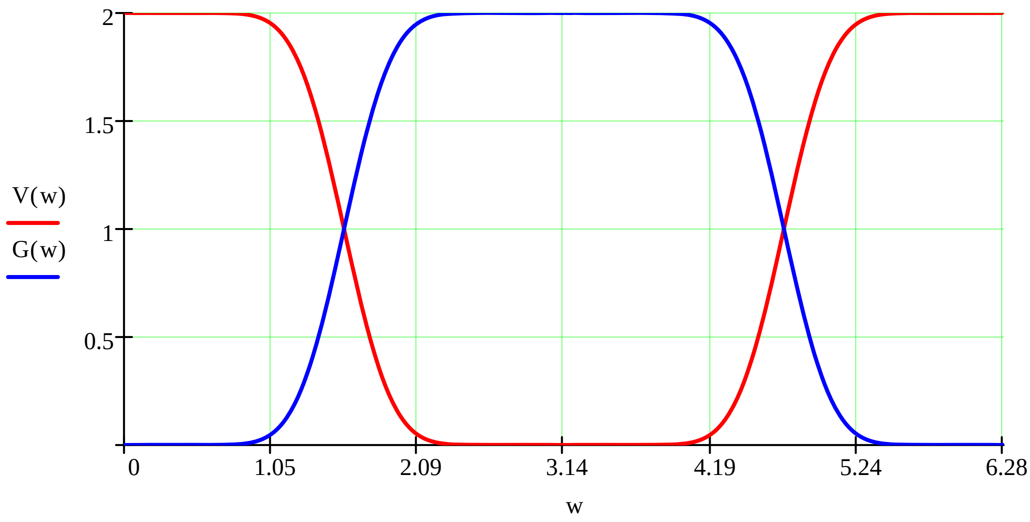

Рис.4 Частотные характеристики фильтров верхнего и нижнего канала

Прямое дискретное вейвлет преобразование

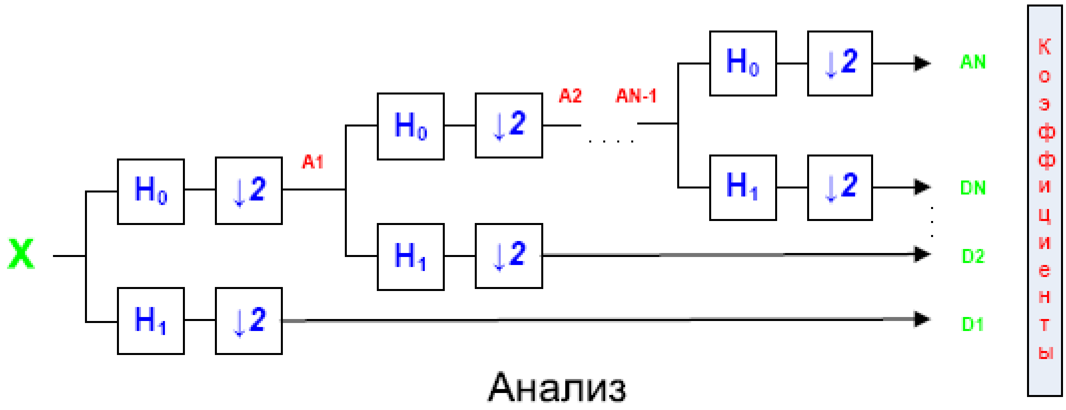

Рис.5 Алгоритм быстрого прямого дискретного вейвлет преобразования

Результатом применения алгоритма быстрого прямого дискретного вейвлет преобразования (Рис.5) к входному сигналу ***X*** на ***N*** -ом уровне будут являться:
- ***N*** - коэффициентов детализации ***D1***, ***D2***,..., ***DN***
- один коэффициент аппроксимации ***AN***, на ***N*** -ом уровне анализа входного сигнала.

Для прямого вейвлет преобразования необходимы фильтры анализа $\mathbf{H_0}$ и $\mathbf{H_1}$:

$$N := \mathrm{last}(H_0) \qquad k := 0..N \qquad N = 13$$

$$H0 := H_0 \qquad\qquad H1_k := (-1)^k \cdot H0_{N-k}$$

H0 =

	0
0	0.07785
1	0.39654
2	0.72913
3	0.46978
4	-0.14391
5	-0.22404
6	0.07131
7	0.08061
8	-0.03803
9	-0.01657
10	0.01255
11	$4.29578 \cdot 10^{-4}$
12	$-1.80164 \cdot 10^{-3}$
13	$3.53714 \cdot 10^{-4}$

H1 =

	0
0	$3.53714 \cdot 10^{-4}$
1	$1.80164 \cdot 10^{-3}$
2	$4.29578 \cdot 10^{-4}$
3	-0.01255
4	-0.01657
5	0.03803
6	0.08061
7	-0.07131
8	-0.22404
9	0.14391
10	0.46978
11	-0.72913
12	0.39654
13	-0.07785

Обратное дискретное вейвлет преобразование

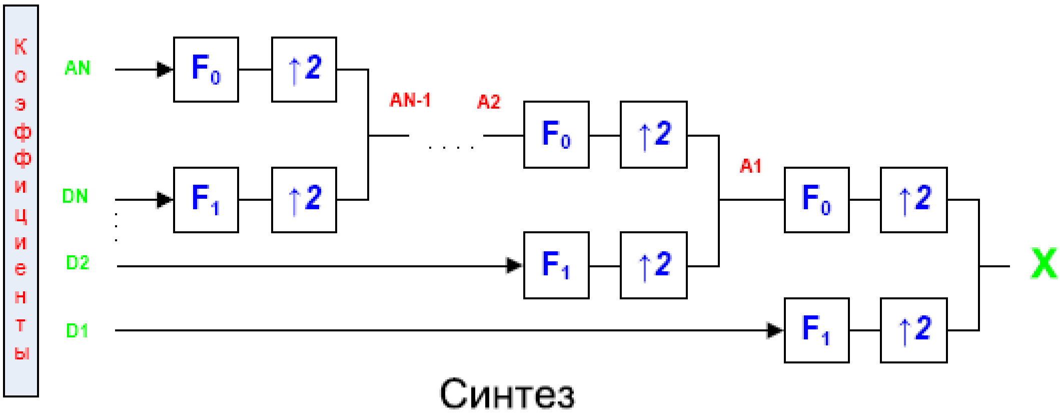

Рис.6 Алгоритм быстрого обратного дискретного вейвлет преобразования

В результате применения алгоритма быстрого обратного дискретного вейвлет преобразования (Рис.6) к коэффициентам ***D1***, ***D2***,..., ***DN*** и ***AN*** - сигнал ***X*** восстановится точно.

Для обратного вейвлет преобразования необходимы фильтры синтеза **F_0** и **F_1**:

$$N := \mathrm{last}(H0) \qquad k := 0..N \qquad N = 13$$

$$F0_k := H0_{N-k} \qquad\qquad F1_k := (-1)^{k+1} \cdot H0_k$$

F0 =

	0
0	$3.53714 \cdot 10^{-4}$
1	$-1.80164 \cdot 10^{-3}$
2	$4.29578 \cdot 10^{-4}$
3	0.01255
4	-0.01657
5	-0.03803
6	0.08061
7	0.07131
8	-0.22404
9	-0.14391
10	0.46978
11	0.72913
12	0.39654
13	0.07785

F1 =

	0
0	-0.07785
1	0.39654
2	-0.72913
3	0.46978
4	0.14391
5	-0.22404
6	-0.07131
7	0.08061
8	0.03803
9	-0.01657
10	-0.01255
11	$4.29578 \cdot 10^{-4}$
12	$1.80164 \cdot 10^{-3}$
13	$3.53714 \cdot 10^{-4}$

Определим процедуры позволяющие осуществить прямое и обратное дискретное вейвлет преобразование, по изложенным выше алгоритмам.

Программный модуль ***Conv(X,Y)*** вспомогательный, он осущесвляет ***линейную дискретную свертку*** двух дискретных сигналов ***X*** и ***Y***.

$$
\mathrm{Conv}(X,Y) \equiv \left|
\begin{array}{l}
Lx \leftarrow \mathrm{last}(X) \\
Ly \leftarrow \mathrm{last}(Y) \\
\text{if } Lx \le Ly \\
\quad \left| \begin{array}{l} Xd \leftarrow X \\ Yd \leftarrow Y \\ Ldx \leftarrow Lx \\ Ldy \leftarrow Ly \end{array} \right. \\
\text{otherwise} \\
\quad \left| \begin{array}{l} Xd \leftarrow Y \\ Yd \leftarrow X \\ Ldx \leftarrow Ly \\ Ldy \leftarrow Lx \end{array} \right. \\
\text{for } k \in 0..Ldx + Ldy \\
\quad \left| \begin{array}{l}
C_k \leftarrow \displaystyle\sum_{i=0}^{k} Xd_i \cdot Yd_{k-i} \quad \text{if } 0 \le k \le Ldx - 1 \\
C_k \leftarrow \displaystyle\sum_{i=0}^{Ldx} Xd_i \cdot Yd_{k-i} \quad \text{if } Ldx \le k \le Ldy \\
C_k \leftarrow \displaystyle\sum_{i=k-Ldy}^{Ldx} Xd_i \cdot Yd_{k-i} \quad \text{otherwise}
\end{array} \right. \\
C
\end{array}
\right.
$$

Программный модуль ***WP(DAT,N)*** вспомогательный.

$$
\mathrm{WP}(DAT,N) \equiv \left|
\begin{array}{l}
LD \leftarrow \mathrm{length}(DAT) \\
A \leftarrow \mathrm{mod}(LD,2) \\
B \leftarrow \mathrm{mod}(N,2) \\
DOUT \leftarrow \mathrm{submatrix}\left(DAT, \dfrac{LD-N}{2}, \dfrac{LD+N}{2} - 1, 0, 0\right) \quad \text{if } A = 0 \wedge B = 0 \\
DOUT \leftarrow \mathrm{submatrix}\left(DAT, \dfrac{LD-N-1}{2}, \dfrac{LD+N-3}{2}, 0, 0\right) \quad \text{if } A = 0 \wedge B = 1 \\
DOUT \leftarrow \mathrm{submatrix}\left(DAT, \dfrac{LD-N-1}{2}, \dfrac{LD+N-3}{2}, 0, 0\right) \quad \text{if } A = 1 \wedge B = 0 \\
DOUT \leftarrow \mathrm{submatrix}\left(DAT, \dfrac{LD-N}{2}, \dfrac{LD+N}{2} - 1, 0, 0\right) \quad \text{otherwise} \\
DOUT
\end{array}
\right.
$$

Программный модуль ***DVP(Data,H0,H1,N)*** выполняет ***N*** уровней прямого дискретного вейвлет преобразования вектора данных ***Data*** с фильтрами анализа ***H0*** и ***H1***.

$$
\mathrm{DVP}(\mathrm{Data},\mathrm{H0},\mathrm{H1},\mathrm{N}) := \left|\begin{array}{l}
m \leftarrow \mathrm{last}(\mathrm{Data})\\
n \leftarrow \mathrm{last}(\mathrm{H0})\\
LL_0 \leftarrow \mathrm{floor}\left(\dfrac{m+n-1}{2}\right)\\
L_0 \leftarrow m\\
\text{for } s \in 0..N-1\\
\quad \left|\begin{array}{l}
LOD \leftarrow \mathrm{Conv}(\mathrm{H0},\mathrm{Data})\\
HID \leftarrow \mathrm{Conv}(\mathrm{H1},\mathrm{Data})\\
\text{for } k \in 0..LL_s\\
\quad \left|\begin{array}{l}
DH_{k,s} \leftarrow LOD_{2k+1}\\
DH_{k,s+N} \leftarrow HID_{2k+1}
\end{array}\right.\\
LL_{s+1} \leftarrow \mathrm{floor}\left(\dfrac{LL_s+n-1}{2}\right)\\
L_{s+1} \leftarrow \mathrm{floor}\left(\dfrac{L_s+n-1}{2}\right)\\
\mathrm{Data} \leftarrow DH^{\langle s\rangle}
\end{array}\right.\\
\begin{pmatrix} DH\\ L \end{pmatrix}
\end{array}\right.
$$

Таким образом, результатом применения программного модуля ***DVP(Data,H0,H1,N)*** будет матрица вида $\begin{pmatrix} DH\\ L \end{pmatrix}$, где:

- ***DH*** - матрица содержащая в первых ***N*** столбцах коэффициенты аппроксимации, и в последних ***N*** столбцах коэффициенты детализации
- ***L*** - матрица содержащая величину длинны коэффициентов вейвлет преобразования на каждом уровне анализа входного сигнала.

Программный модуль ***OVP(CF,F0,F1,Prg,Tp)*** выполняет обратное дискретное вейвлет преобразование с использованием матрицы ***CF*** - коэффициентов детализации и аппроксимации, полученных в результате прямого вейвлет преобразования (***CF=DVP(Data,H0,H1,N)***) и фильтров синтеза ***F0*** и ***F1***.

Tp - определяет форму порога и может принимать только два значения:
"***H***" - жесткий порог
"***S***" - мягкий порог

Prg - порог, ниже которого все детализирующие коэффициенты обнуляются, должен быть числом большим нуля. Если ***Prg=0***, то детализирующие коэффициенты не изменяются, значение ***Tp*** приэтом может быть либо "***H***" или "***S***" .

Отобразим передаточные функции жесткого (Рис.7) и мягкого порога (Рис.8)

$$\mathrm{Prg} := 5$$

$$H(x,\mathrm{Prg}) := \mathrm{if}(|x| \ge \mathrm{Prg}, x, 0)$$

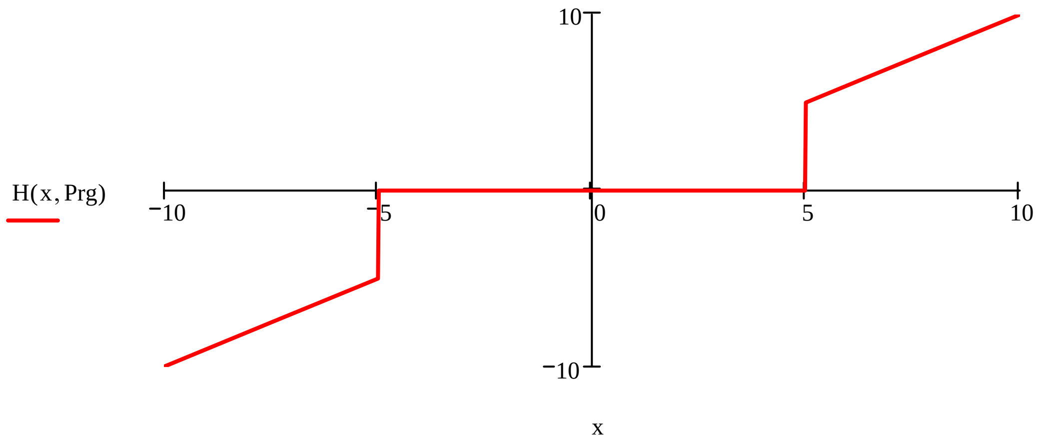

Рис.7 Передаточная функция жесткого порога, при $\mathrm{Prg} = 5$

$$S(x,\mathrm{Prg}) := \mathrm{if}[|x| \ge \mathrm{Prg}, \mathrm{sign}(x)\cdot(|x| - \mathrm{Prg}), 0]$$

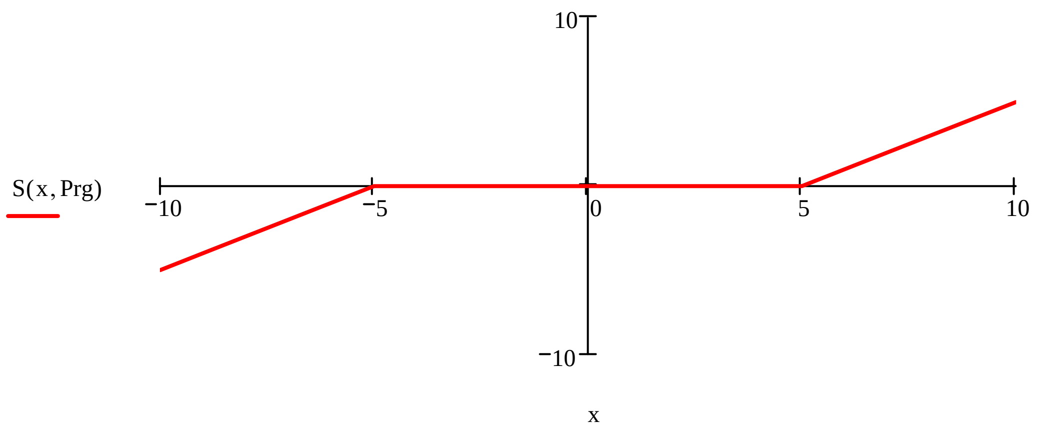

Рис.8 Передаточная функция мягкого порога, при $\mathrm{Prg} = 5$

$$
\mathrm{OVP}(CF, F0, F1, Prg, Tp) := \left|
\begin{array}{l}
D \leftarrow CF_{0,0} \\
L \leftarrow CF_{1,0} \\
m \leftarrow \mathrm{cols}(D) \\
n \leftarrow \mathrm{rows}(D) \\
N \leftarrow \dfrac{m}{2} \\
\text{for } \; j \in N..m-1 \\
\quad \text{for } \; i \in 0..n-1 \\
\qquad \left| \begin{array}{l}
DH_{i,j} \leftarrow D_{i,j} \;\; \text{if } \left(\left|D_{i,j}\right| \ge Prg\right) \wedge Tp = \text{"H"} \\
DH_{i,j} \leftarrow \mathrm{sign}\left(D_{i,j}\right)\cdot\left(\left|D_{i,j}\right| - Prg\right) \;\; \text{if } \left(\left|D_{i,j}\right| \ge Prg\right) \wedge Tp = \text{"S"} \\
DH_{i,j} \leftarrow 0 \;\; \text{otherwise}
\end{array} \right. \\
APR \leftarrow D^{\langle N-1 \rangle} \\
DT \leftarrow \left(\mathrm{reverse}\left((DH)^{T}\right)\right)^{T} \\
\text{for } \; t \in 0..N-1 \\
\quad \left| \begin{array}{l}
\text{for } \; s \in 0..2\cdot L_{N-t} + 2 \\
\quad \left| \begin{array}{l}
DR_{s,t} \leftarrow 0 \\
AR_{s} \leftarrow 0
\end{array} \right. \\
\text{for } \; s \in 0..L_{N-t} \\
\quad \left| \begin{array}{l}
DR_{2s+1,t} \leftarrow DT_{s,t} \\
AR_{2s+1} \leftarrow APR_{s}
\end{array} \right. \\
Rec^{\langle t \rangle} \leftarrow \mathrm{Conv}(F0, AR) + \mathrm{Conv}\left(F1, DR^{\langle t \rangle}\right) \\
Rc^{\langle t \rangle} \leftarrow \mathrm{WP}\left(Rec^{\langle t \rangle}, L_{N-t-1} + 1\right) \\
APR \leftarrow Rc^{\langle t \rangle}
\end{array} \right. \\
APR
\end{array}
\right.
$$

Проверка тождественности функций прямого ***DVP(Data,H0,H1,N)*** и обратного ***OVP(CF,F0,F1,Prg,Tp)*** вейвлет преобразования

1) Задаем параметры синусоиды

$$i := 0..1000$$

$$DXD_i := \sin\left(\frac{4\cdot\pi}{256} i\right)$$

2) Задаем шумовой сигнал
- m=0 - значение математического ожидания
- σ =0.3 - значение среднего квадратического отклонения.

$$m := 0 \qquad \sigma := 0.3$$

$$Shum := rnorm(length(DXD), m, \sigma)$$

3) Создаем тестовый сигнал - суммируя значения синусоиды и шума

$$DAT_i := DXD_i + Shum_i$$

4) Выполняем N=5 уровней прямого вейвлет преобразования

$$N := 5 \qquad WDB := DVP(DAT, H0, H1, N)$$

5) Выполняем обратное вейвлет преобразование

$$IWD := OVP(WDB, F0, F1, 0, \text{"H"})$$

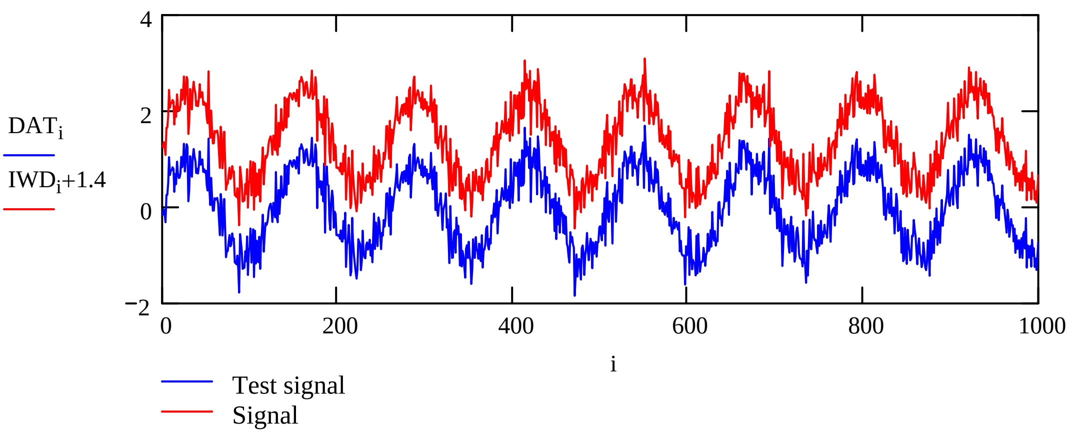

Рис.9 Тестовый сигнал и сигнал восстановленный по коэффициентам полученным в результате прямого дискретного вейвлет преобразования тестого сигнала

6) Квадратичное отклонение сигнала восстановленного после вейвлет преобразования от тестого сигнала

$$\sum_{i=0}^{last(DAT)} \left[\left(\left|DAT_i - IWD_i\right|\right)^2\right] = 0$$

Как видно, сигнал восстановлен точно по коэффициентам полученным в результате прямого дискретного вейвлет преобразования.

Сравнение жесткого и мягкого порога при синтезе сигнала по коэффициентам полученным в результате прямого дискретного вейвлет преобразования.

1) Задание количества уровней прямого вейвлет преобразования

$$N := 5$$

2) Вычисление $N = 5$ уровней прямого вейвлет преобразования

$$WDB := DVP(DAT, H0, H1, N)$$

3) Задание значения порога

$$Prg := 2\sigma$$

4) Вычисление обратного вейвлет преобразование с жестким и мягким видом порога, при значении порога $Prg = 0.6$ (Рис.10):

$IWH := OVP(WDB, F0, F1, Prg, "H")$ $\qquad$ $IWS := OVP(WDB, F0, F1, Prg, "S")$

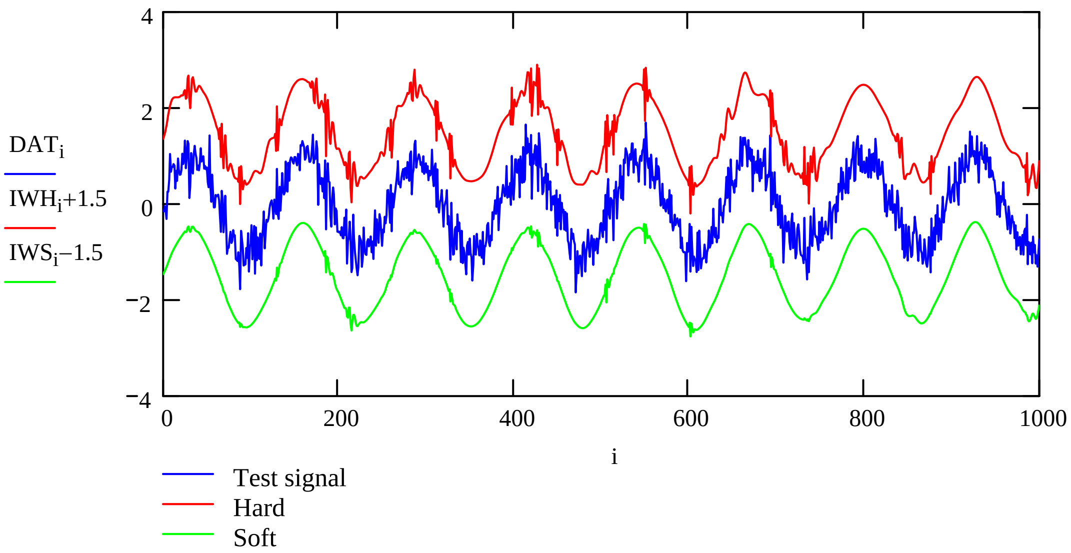

Рис.10 Тестовый сигнал и сигнал восстановленный по коэффициентам полученным в результате прямого дискретного вейвлет преобразования тестого сигнала, с функцией жесткого и мягкого порога.

Для практических приложений интерес представляют модификации вейвлет разложения, а не тождественные преобразования. Например, для сжатия важно представить сигнал небольшим числом коэффициентов прямого вейвлет преобразования.

Программный модуль ***Cmpr(CF,K)*** выделяет в матрице ***CF*** - коэффициентов детализации и аппроксимации, полученных в результате прямого вейвлет преобразования ***CF=DVP(Data,H0,H1,N)***, только ***K*** наибольших по абсолютному значению коэффициентов детализации и аппроксимации, полагая остальные равными нулю.

$$
\mathrm{Cmpr}(CF,K) := \left|
\begin{array}{l}
Data \leftarrow CF_{0,0} \\
m \leftarrow \mathrm{cols}(Data) \\
n \leftarrow \mathrm{rows}(Data) \\
N \leftarrow \dfrac{m}{2} \\
DD \leftarrow \mathrm{submatrix}(Data, 0, n-1, N-1, m-1) \\
md \leftarrow \mathrm{cols}(DD) \\
DST \leftarrow DD^{\langle 0 \rangle} \\
\text{for } j \in 1 .. md-1 \\
\quad DST \leftarrow \mathrm{stack}\left(DST, DD^{\langle j \rangle}\right) \\
Dsort \leftarrow \mathrm{sort}\left(\overrightarrow{|DST|}\right) \\
Nx \leftarrow \mathrm{last}(Dsort) \\
A \leftarrow Dsort_{Nx-K} \\
\text{for } j \in N-1 .. m-1 \\
\quad \text{for } i \in 0 .. n-1 \\
\quad\quad \left| \begin{array}{l} DH_{i,j} \leftarrow Data_{i,j} \text{ if } \left|Data_{i,j}\right| > A \\ DH_{i,j} \leftarrow 0 \text{ otherwise} \end{array} \right. \\
\begin{pmatrix} DH \\ CF_{1,0} \end{pmatrix}
\end{array}
\right.
$$

Применение функции *Cmpr*(*CF,K*) для сжатия тестового сигнала

1) Задание тестового сигнала

$$i := 0 .. 1000$$

$$DXD_i := \sin\left[2\cdot\pi\left(\frac{i}{500}\right)^2\right]$$

2) Задание количества уровней прямого вейвлет преобразования тестового сигнала

$$N := 6$$

3) Вычисление $N = 6$ уровней прямого вейвлет преобразования

$$WDB := DVP(DXD, H0, H1, N)$$

4) Задание количества наибольших по абсолютному значению коэффициентов ***K***

$$K := 16$$

5) Выделение из матрицы коэффициентов детализации и аппроксимации, полученных в результате прямого вейвлет преобразования только $K = 16$ наибольших по абсолютному значению коэффициентов, с помощью функции ***Cmpr(CF,K)***

$$WDC := Cmpr(WDB, K)$$

6) Расчет обратного вейвлет преобразование по $K = 16$ коэффициентам и отображение полученного результата (Рис.11)

$$IWD := OVP(WDC, F0, F1, 0, "H")$$

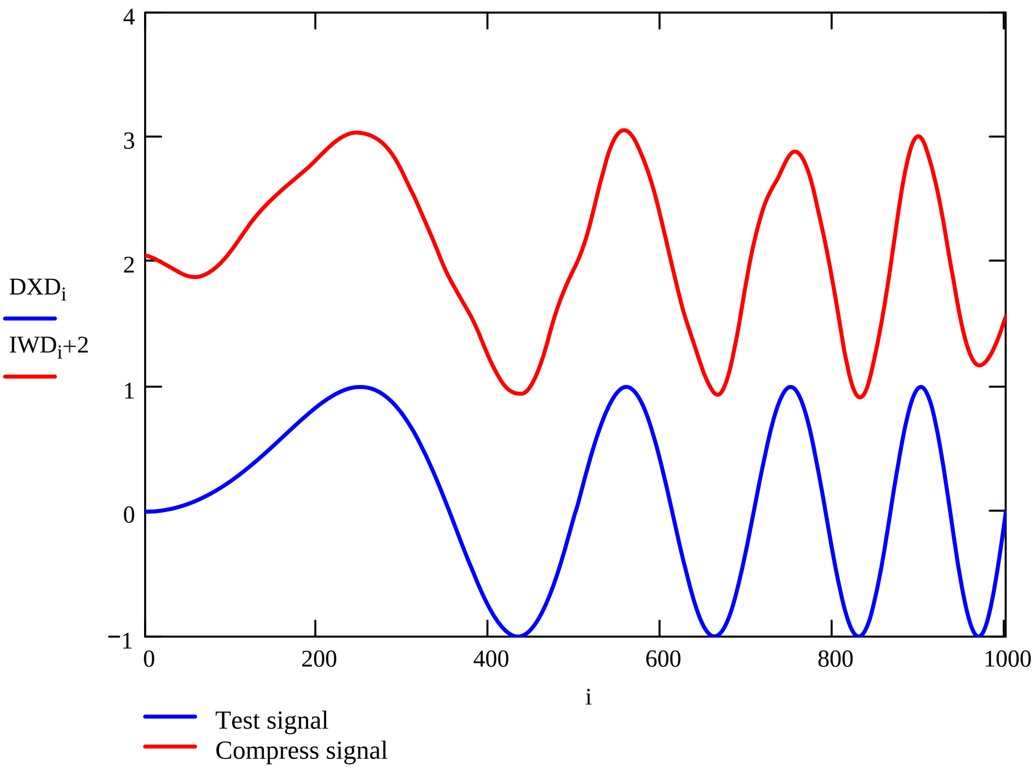

Рис.11 Тестовый сигнал и сигнал восстановленный по $K = 16$ наибольшим по абсолютному значению коэффициентам аппроксимации и детализации прямого вейвлет преобразования тестового сигнала

Базисные Функции Вейвлет Преобразования

В результате синтеза, сигнал представляется суммой базисных функций вейвлет преобразования с весами, равными коэффициентам вейвлет разложения. Следовательно, чтобы получить единичную базисную функцию, мы должны подать на блок синтеза серию коэффициентов, равных нулю, за исключением одного, равного единице.

Программный модуль ***VidWave(X,Nv)*** выполняет ***расчет базисных функций*** вейвлет преобразования.

- значение ***X*** равное ***F0*** приводит к масшатабирующей функции
- значение ***X*** равное ***F1*** приводит к вейвлет функции

Переменная ***Nv*** определяет количество итераций для построения базисной функции.

$$\mathrm{VidWave}(X, Nv) := \left| \begin{array}{l} \text{for } k \in 1..Nv \\ \quad \left| \begin{array}{l} \text{if } k > 1 \\ \quad \left| \begin{array}{l} Lx \leftarrow \mathrm{length}(X) \\ \text{for } n \in 0..Lx - 1 \\ \quad Y_{2n+1} \leftarrow X_n \\ X \leftarrow \sqrt{2}\cdot \mathrm{Conv}(Y, F0) \end{array} \right. \\ X \leftarrow X \quad \text{otherwise} \end{array} \right. \\ X \end{array} \right.$$

1) Задаем количество итераций для построения базисных функций

$$Nv := 5$$

2) Рассчитываем базисную масштабирующую функцию и строим ее график (Рис.12)

$$Sc := \mathrm{VidWave}(F0, Nv) \qquad i := 0..\mathrm{last}(Sc)$$

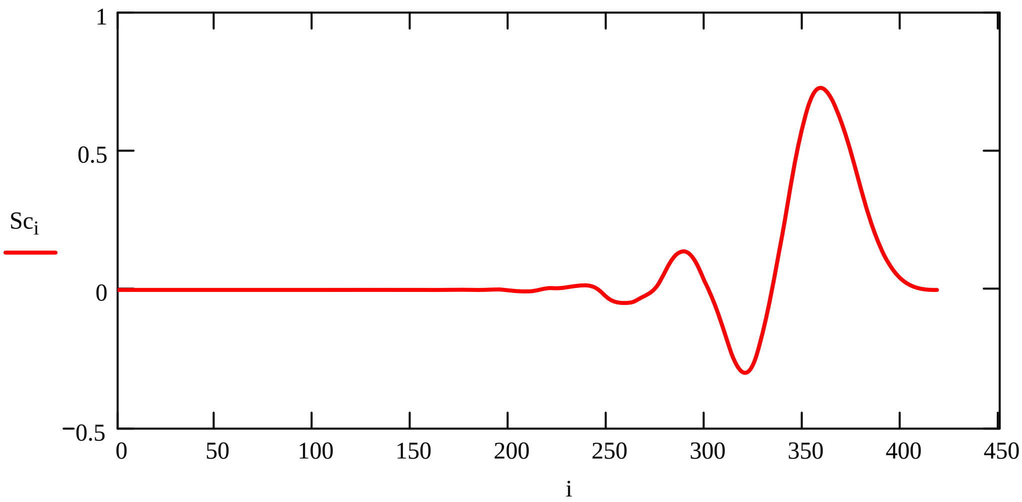

Рис.12 Базисная масштабирующая функция

3) Рассчитываем базисную вейвлет функцию и строим ее график (Рис.13)

$$Wv := VidWave(F1, Nv) \quad i := 0..last(Wv)$$

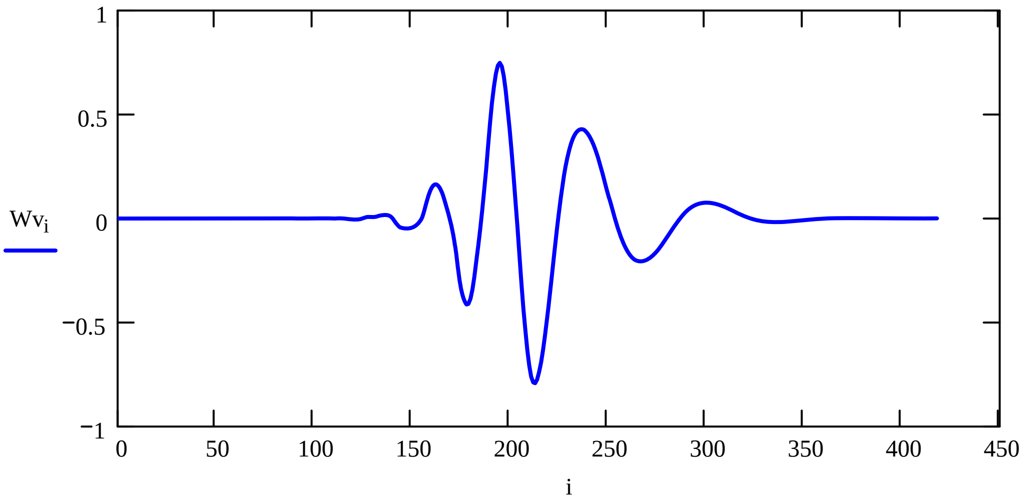

Рис.13 Базисная вейвлет функция