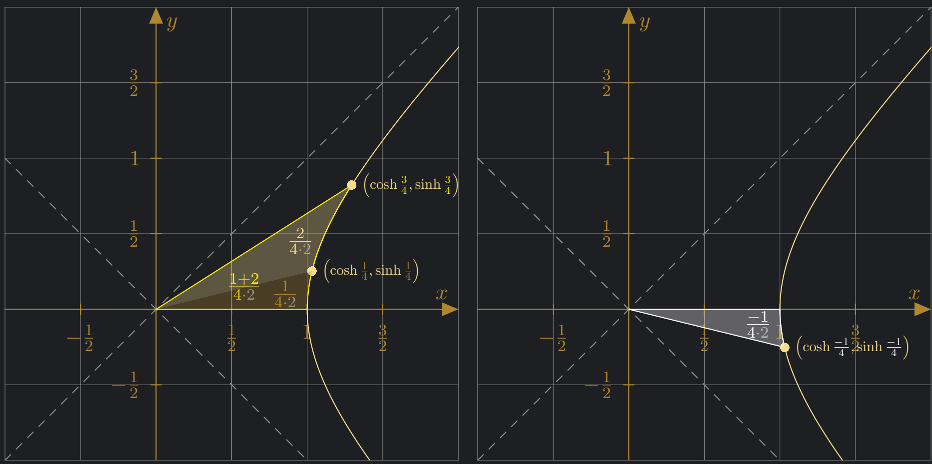
1 Tryb hiperboliczny

Przypomnijmy definicje funkcji hiperbolicznych

$$\sinh\alpha = \frac{e^\alpha - e^{-\alpha}}{2} \qquad \cosh\alpha = \frac{e^\alpha + e^{-\alpha}}{2} \qquad \operatorname{tgh}\alpha = \frac{e^\alpha - e^{-\alpha}}{e^\alpha + e^{-\alpha}} \qquad \operatorname{ctgh}\alpha = \frac{e^\alpha + e^{-\alpha}}{e^\alpha - e^{-\alpha}}$$

Warto zaznaczyć, że argumenty tych funkcji nie są kątami. Można nadać im sens geometryczny, mianowicie, że argument określa dwukrotność pola powierzchni ograniczonej dodatnią osią x, hiperbolą oraz promieniem wodzącym punktu na hiperboli.



Z tym, że to pole jest brane ze znakiem, czyli jest dodatnie gdy $y > 0$ i ujemne gdy $y < 0$. Mamy w zasadzie pseudopole, choć dalej dla wygody i tak będę mówić krótko pole.

Wprost z tych definicji wynikają wzory

$$\begin{aligned} \sinh(\alpha+\beta) &= \sinh\alpha\cosh\beta + \cosh\alpha\sinh\beta \\ \cosh(\alpha+\beta) &= \cosh\alpha\cosh\beta + \sinh\alpha\sinh\beta \end{aligned}$$

Te nas prowadzą do zapisu macierzowego

$$\begin{bmatrix} \cosh(\alpha+\beta) \\ \sinh(\alpha+\beta) \end{bmatrix} = \begin{bmatrix} \cosh\beta & \sinh\beta \\ \sinh\beta & \cosh\beta \end{bmatrix} \cdot \begin{bmatrix} \cosh\alpha \\ \sinh\alpha \end{bmatrix}$$

Skoro punkt na hiperboli ma współrzędne $(\cosh\delta, \sinh\delta)$, to niech punkt $A = (\cosh\alpha, \sinh\alpha)$, wówczas punkt po przemieszczaniu na hiperboli o dodatkowe zakreślone pseudo pole $\frac{\beta}{2}$ stanie się punktem $A' = (\cosh(\alpha+\beta), \sinh(\alpha+\beta))$.

$$\begin{bmatrix} A'_x \\ A'_y \end{bmatrix} = \begin{bmatrix} \cosh\beta & \sinh\beta \\ \sinh\beta & \cosh\beta \end{bmatrix} \cdot \begin{bmatrix} A_x \\ A_y \end{bmatrix}$$

1.1 Przekształcenie

Możemy to równanie przekształcić

$$\begin{cases} A'_x = A_x\cosh\varphi + A_y\sinh\varphi \\ A'_y = A_y\cosh\varphi + A_x\sinh\varphi \end{cases} \Rightarrow \begin{cases} A'_x = (A_x + A_y\operatorname{tgh}\varphi)\cosh\varphi \\ A'_y = (A_y + A_x\operatorname{tgh}\varphi)\cosh\varphi \end{cases}$$

Z definicji funkcji hiperbolicznych wynika wzór $\cosh^2 x - \sinh^2 x = 1$. Wzór ten nazywamy jedynka hiperboliczna. Wykorzystując go dostajemy analogiczną tożsamość jak w trybie okrężnym.

$$\cosh\varphi = \frac{1}{\frac{1}{\cosh\varphi}} = \frac{1}{\sqrt{\frac{1}{\cosh^2\varphi}}} = \frac{1}{\sqrt{\frac{\cosh^2\varphi - \sinh^2\varphi}{\cosh^2\varphi}}} = \frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi}}$$

$$\begin{cases} A'_x = (A_x + A_y\operatorname{tgh}\varphi)\cosh\varphi \\ A'_y = (A_y + A_x\operatorname{tgh}\varphi)\cosh\varphi \end{cases} \Rightarrow \begin{cases} A'_x = (A_x + A_y\operatorname{tgh}\varphi)\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi}} \\ A'_y = (A_y + A_x\operatorname{tgh}\varphi)\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi}} \end{cases}$$

1.2 Przesunięcia na hiperboli jako suma mniejszych przesunięć

Podobnie jak w trybie okrężnym mamy

$$\varphi = \varphi_0 + \varphi_1 + \ldots + \varphi_{n-1}$$

Otrzymujemy układ rekurencji.

$$\begin{cases} \tilde{x}_{k+1} = (\tilde{x}_k + \tilde{y}_k \operatorname{tgh} \varphi_k) \frac{1}{\sqrt{1-\operatorname{tgh}^2 \varphi_k}} \\ \tilde{y}_{k+1} = (\tilde{y}_k + \tilde{x}_k \operatorname{tgh} \varphi_k) \frac{1}{\sqrt{1-\operatorname{tgh}^2 \varphi_k}} \\ \tilde{x}_0 = A_x \\ \tilde{y}_0 = A_y \end{cases}$$

Po n krokach otrzymujemy dobre przybliżenie obrazu $A' \approx (\tilde{x}_n, \tilde{y}_n)$. Zauważmy, że funkcja area tangens hiperboliczny jest funkcją nieparzystą, tzn. $\operatorname{artgh}(-x) = -\operatorname{artgh} x$. Zatem znając wartość artgh dla pewnego argumentu znamy też wartość tej funkcji dla argumentu przeciwnego. Wystarczy zmienić znak na przeciwny. Rozważnym zbiór γ_k analogiczny jak w trybie okrężnym. Zauważmy, iż $\operatorname{artgh} 2^0 = \operatorname{artgh} 1$ nie nie istniej. Pierwszą wartością pola jest $\operatorname{artgh} 2^{-1}$.

$$\left\{ \gamma_k \in (0; 0,55) : \gamma_k = \operatorname{artgh} 2^{-(k+1)} \quad \wedge \quad k \in \mathbb{N} \right\}$$

Ponownie można raz wyznaczyć wartości tych kątów dla niekoniecznie dużego n np. $n = 40$ i zapamiętać.

k	0	1	2	3	4	5	6	7	...
γ_k	$\operatorname{artgh} \frac{1}{2}$	$\operatorname{artgh} \frac{1}{4}$	$\operatorname{artgh} \frac{1}{8}$	$\operatorname{artgh} \frac{1}{16}$	$\operatorname{artgh} \frac{1}{32}$	$\operatorname{artgh} \frac{1}{64}$	$\operatorname{artgh} \frac{1}{128}$	$\operatorname{artgh} \frac{1}{256}$	...
γ_k	0,5493	0,2554	0,1257	0,0626	0,0313	0,0156	0,0078	0,0039	...

Uwaga γ_k nie jest kątem, tylko podwojoną wartością zakreślonego pseudopola, wynikającego z promienia wodzącego, punktu na hiperboli. Jak zawsze bieżący φ_k jest równy γ_k z ewentualnie zmienionym znakiem.

$$\varphi_k = \epsilon_k \gamma_k \quad \text{gdzie} \quad \epsilon_k \{-1, 1\}$$

Pojawiają się jednak problemy kolejne wartości maleją bardziej, niż o 2 razy i to w każdym kroku. To prowadzi do tego, że niektóre wartości φ nie da się rozpisać na sumę ciągu lub szeregu wyrazów, które są $\pm\gamma_k$. Krótko mówiąc nie ma zbieżności. Aby to ograniczenie obejść utarł się pomysł, aby niektóre iteracje powtarzać. Okazuje się, że jeśli powtórzy się iteracje $4, 13, 40, 121, \ldots, k, 3k+1, \ldots$, to wówczas, każda wartość w przybliżonym przedziale od $-1,11$ do $1,11$ uda się rozpisać na sumę ciągu lub szeregu tych γ_k branych z plusem lub minusem. Pozostawiam to bez dowodu. Wprowadza to pewną komplikację, ale rozwiązuje problem. Trzeba uważać co teraz nazywamy iteracją oraz komplikuje opis matematyczny. Dla ustalenia uwagi pominę na razie te podwójne iteracje. Wyznaczone zależności zmodyfikujemy na koniec.

Znaki znajdujemy analogicznie do poprzednich trybów. Jeśli aktualnie naliczona suma pseudo pól, tzn. $\varphi_0 + \varphi_1 + \ldots + \varphi_k$ jest mniejsza od szukanego pseudo pola φ to następne pole γ_{k+1} trzeba dodać. Jak aktualne naliczona suma pseudo pól, jest większa od szukanego pseudo pola φ to następne pole γ_{k+1} trzeba odjąć.

Tę zasadę można odwrócić i szukane φ z kroku na krok modyfikować, aż będzie dostatecznie blisko 0 lub gdy uznamy, że osiągnęliśmy maksymalną liczbę kroków n. Niech β_k oznacza argument, po kroku k, jaki pozostał jeszcze do przesunięcia po hiperboli, aby znaleźć się w punkcie A', będącym obrazem punkt A.

$$\begin{cases} \beta_{k+1} = \beta_k - \varphi_k \\ \beta_0 = \varphi \end{cases}$$

Po n krokach $\beta_n \approx 0$. Zauważmy także, że $\varphi_k = \epsilon_k \gamma_k = \operatorname{sgn}(\beta_k)\gamma_k$, a co za tym idzie

$$\begin{cases} \beta_{k+1} = \beta_k - \operatorname{sgn}(\beta_k)\gamma_k \\ \beta_0 = \varphi \end{cases}$$

$$\operatorname{tgh}\varphi_k = \operatorname{tgh}(\epsilon_k\gamma_k) = \epsilon_k \operatorname{tgh}\gamma_k = \operatorname{sgn}(\beta_k)\operatorname{tgh}\gamma_k = \operatorname{sgn}(\beta_k)2^{-(k+1)}$$

W ten sposób problem mnożenia $\tilde{y}_k \cdot \operatorname{tgh}\varphi_k$ oraz $\tilde{x}_k \cdot \operatorname{tgh}\varphi_k$ został rozwiązany. Pozostała kwestia mnożenia przez $\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}}$.

$$\begin{cases} \tilde{x}_{k+1} = (\tilde{x}_k + \tilde{y}_k \operatorname{tgh}\varphi_k)\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \\ \tilde{y}_{k+1} = (\tilde{y}_k + \tilde{x}_k \operatorname{tgh}\varphi_k)\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \\ \tilde{x}_0 = A_x \\ \tilde{y}_0 = A_y \end{cases} \rightarrow \begin{cases} \tilde{x}_{k+1} = \left(\tilde{x}_k + \operatorname{sgn}(\beta_k)\tilde{y}_k \cdot 2^{-(k+1)}\right)\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \\ \tilde{y}_{k+1} = \left(\tilde{y}_k + \operatorname{sgn}(\beta_k)\tilde{x}_k \cdot 2^{-(k+1)}\right)\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \\ \tilde{x}_0 = A_x \\ \tilde{y}_0 = A_y \end{cases}$$

1.3 Mnożenie przez $\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}}$

Przyglądając się podobnie dwóm kolejnym iteracją dochodzimy do analogicznego wniosku jak w trybie okrężnym. Mianowicie zamiast przemnażać w każdej iteracji przez $\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}}$ można na koniec przemnożyć przez iloczyn tych wszystkich wyrażeń.

$$\begin{cases} \tilde{x}_{k+1} = (\tilde{x}_k + \tilde{y}_k \operatorname{tgh}\varphi_k)\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \\ \tilde{y}_{k+1} = (\tilde{y}_k + \tilde{x}_k \operatorname{tgh}\varphi_k)\frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \\ \tilde{x}_0 = A_x \\ \tilde{y}_0 = A_y \end{cases} \rightarrow \begin{cases} x_{k+1} = x_k + y_k \operatorname{tgh}\varphi_k \\ y_{k+1} = y_k + x_k \operatorname{tgh}\varphi_k \\ x_0 = A_x \\ y_0 = A_y \end{cases}$$

Po n iteracjach otrzymamy x_n i y_n, które mnożymy przez dobraną stałą

$$\begin{cases} \tilde{x}_n = x_n \prod_{k=0}^{n-1} \frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \\ \tilde{y}_n = y_n \prod_{k=0}^{n-1} \frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \end{cases}$$

Natomiast podobnie zamiast przemnażać na końcu można przemnożyć na początku. Od tego momentu zmienne akcentowane tyldą nie są już potrzebne.

$$\begin{cases} x_{k+1} = x_k + y_k \operatorname{tgh}\varphi_k \\ y_{k+1} = y_k + x_k \operatorname{tgh}\varphi_k \\ x_0 = A_x \prod_{k=0}^{n-1} \frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \\ y_0 = A_y \prod_{k=0}^{n-1} \frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} \end{cases}$$

Zauważmy iż $\operatorname{tgh}^2\gamma_k = \operatorname{tgh}^2(-\gamma_k) = \operatorname{tgh}^2\varphi_k$, oznacza to, że wszystkie $\operatorname{tgh}^2\varphi_k$ są znane z góry, bo znane są z góry wartości pól γ_k.
Rozpatrzmy ciąg $\bar{H} : \mathbb{N}^+ \rightarrow \mathbb{R}$, pierwszy wyraz jest o indeksie 1. Dla odróżnienia ze stałymi w trybie rotacyjnym nazwałem go przez $\bar{H}$. Ciąg jest dany wzorem

$$\bar{H}_n = \prod_{k=0}^{n-1} \frac{1}{\sqrt{1-\operatorname{tgh}^2\varphi_k}} = \prod_{k=0}^{n-1} \frac{1}{\sqrt{1-2^{-2(k+1)}}} = \prod_{k=1}^{n} \frac{1}{\sqrt{1-2^{-2k}}}$$

Nie przyzwyczajaj się do niego, bo za chwilę go zmienimy .

Na daną chwilę nasz algorytm wygląda następująco

$$\begin{cases} x_{k+1} = x_k + \operatorname{sgn}(\beta_k)y_k 2^{-(k+1)} \\ y_{k+1} = y_k + \operatorname{sgn}(\beta_k)x_k 2^{-(k+1)} \\ \beta_{k+1} = \beta_k - \operatorname{sgn}(\beta_k)\gamma_k \\ x_0 = A_x\bar{H}_n \\ y_0 = A_y\bar{H}_n \\ \beta_0 = \varphi \end{cases}$$

1.4 Komplikacja wzorów ze względu na niektóre podwójne iteracje

Podwojenie niektórych iteracji komplikuje opis matematyczny. Modyfikacji ulega także ciąg $\bar{H}_n$, który zmienia się w H_n.

$$\begin{cases} x_{k+1} = x_k + y_k \operatorname{tgh}\varphi_k \\ y_{k+1} = y_k + x_k \operatorname{tgh}\varphi_k \\ \beta_{k+1} = \beta_k - \varphi_k \\ x_0 = A_x \bar{H}_n \\ y_0 = A_y \bar{H}_n \\ \beta_0 = \varphi \end{cases} \quad \longrightarrow \quad \begin{cases} \begin{cases} \hat{x}_{j+1} = \hat{x}_j + \hat{y}_j \operatorname{sgn}\left(\hat{\beta}_j\right) 2^{-(k+1)} \\ \hat{y}_{j+1} = \hat{y}_j + \hat{x}_j \operatorname{sgn}\left(\hat{\beta}_j\right) 2^{-(k+1)} \\ \hat{\beta}_{j+1} = \hat{\beta}_j - \operatorname{sgn}\left(\hat{\beta}_j\right) \gamma_k \\ \hat{x}_0 = x_k \\ \hat{y}_0 = y_k \\ \hat{\beta}_0 = \beta_k \end{cases} \\ t_k = \begin{cases} 2, & \text{dla } k+1 = 4, 13, 40, 121, \ldots, s, 3s+1, \ldots \\ 1, & \text{dla pozostałych } k+1 \end{cases} \\ x_{k+1} = \hat{x}_{t_k} \\ y_{k+1} = \hat{y}_{t_k} \\ \beta_{k+1} = \hat{\beta}_{t_k} \\ x_0 = A_x H_n \\ y_0 = A_y H_n \\ \beta_0 = \varphi \end{cases}$$

Po tym zabiegu przedział zbieżności to $(-1,11; 1,11)$, wartości wynikają z

$$\sum_{k=0}^{\infty} \sum_{j=1}^{t_k} \operatorname{artgh} 2^{-(k+1)} \approx 1,11$$

gdzie

$$t_k = \begin{cases} 2, & \text{dla } k+1 = 4, 13, 40, 121, \ldots, s, 3s+1, \ldots \\ 1, & \text{dla pozostałych } k+1 \end{cases}$$

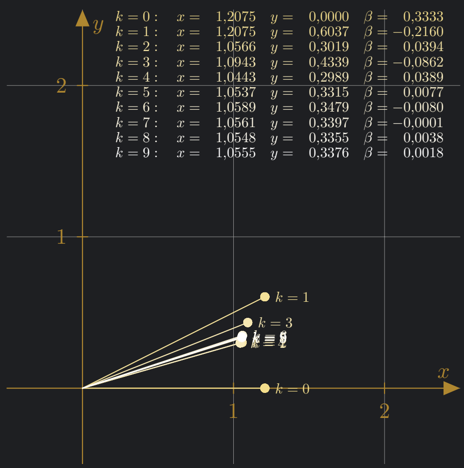
Rozpatrzmy ciąg $H : \mathbb{N}^+ \to \mathbb{R}$, pierwszy wyraz jest o indeksie 1. Dla odróżnienia ze stałymi w trybie okrężnym nazwałem go przez H. Ciąg jest dany wzorem

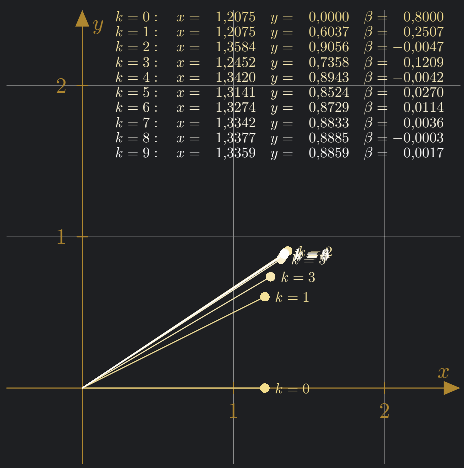
$$H_n = \prod_{k=0}^{n-1} \prod_{j=1}^{t_k} \frac{1}{\sqrt{1 - 2^{-2(k+1)}}}, \qquad \text{gdzie} \quad t_k = \begin{cases} 2, & \text{dla } k+1 = 4, 13, 40, 121, \ldots, s, 3s+1, \ldots \\ 1, & \text{dla pozostałych } k+1 \end{cases}$$

n	1	2	3	4	5	6	7	8	...
H_n	1,1547	1,1926	1,2020	1,2067	1,2073	1,2074	1,2075	1,2075	...

1.5 Przykłady

W pierwszym przykładzie jest obliczany $\sinh \frac{1}{3}$ oraz $\cosh \frac{1}{3}$. W drugim przykładzie $\sinh \frac{4}{5}$ oraz $\cosh \frac{4}{5}$





1.6 Zwiększenie przedziału zbieżności

Pojawia się kolejny problem. Jak powiększyć przedział zbieżności $\varphi \in (-1,11; 1,11)$. Tu w przeciwieństwie do trybu okrężnego, nie dysponujemy wzorami redukcyjnymi bo funkcje hiperboliczne nie są okresowe, dopiero ich odpowiednik zespolone są okresowe i mają okres urojony, ale nas interesują przypadki rzeczywiste. Tu w przeciwieństwie do trybu liniowego samo wykorzystanie własności liczb zmiennoprzecinkowych nie pomoże.

Jednym z pomysłów jest wykorzystanie tożsamości hiperbolicznych

$$\sinh 2x = 2 \sinh x \cosh x$$

$$\cosh 2x = \cosh^2 x \sinh^2 x$$

Mianowicie jeśli chcemy obliczyć $\sinh 3,6$ lub $\cosh 3,6$ to najpierw obliczamy $\sinh 0,9$ oraz $\cosh 0,9$. Następnie wykorzystując podane wzory znajdujemy $\sinh 1,8$ oraz $\cosh 1,8$, a następnie znajdujemy $\sinh 3,6$ oraz $\cosh 3,6$. Zauważmy, że ten tryb wymaga jednak mnożenia i kilku instancji algorytmu lub przełączania

1.7 Tryb hiperboliczny wektorowy

Ten tryb pozwala na obliczenie area tangens hiperbolicznego. Jak nie trudno się domyślić na podstawie doświadczeń z poprzednich trybów. Należy sprowadzić punkt A do punktu A', który będzie leżał na osi x. Zaczynamy od dowolnego punktu na płaszczyźnie $A = (A_x, A_y)$. Chcemy znaleźć dla niego $\varphi = \operatorname{artgh} \frac{A_y}{A_x}$, które przesunie punkt po hiberboli na oś x. Zauważmy jednak, iż dowolny punkt raczej nie leży na hiperboli jednostkowej.

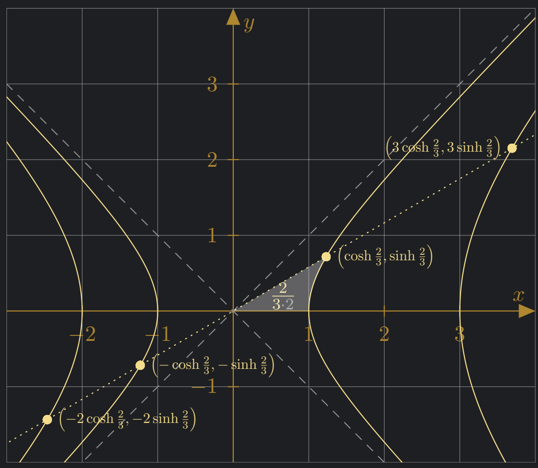
Rozumowanie możemy prowadzić na tej hiperboli na, której znajduje się ten punkt.

$$x^2 - y^2 = a^2$$

Albo przeliczyć ten punkt na adekwatny punkt na hiperboli jednostkowej.

$$\left(\frac{x}{a}\right)^2 - \left(\frac{y}{a}\right)^2 = 1$$

Wybierzmy wersje drugą



Po przesunięciach współrzędna x będzie równa $A'_x = \sqrt{A_x^2 - A_y^2}$. Wyjdźmy od postaci, którą już wyprowadziliśmy. Pomijam chwilowo dla uproszczenia podwójne iteracje.

$$\begin{cases} x_{k+1} = x_k + y_k \operatorname{tgh} \varphi_k \\ y_{k+1} = y_k + x_k \operatorname{tgh} \varphi_k \\ \beta_{k+1} = \beta_k - \varphi_k \end{cases}$$

φ_k określa jak zawsze wykonane przesunięcie w $k+1$ iteracji. Niech β_k będzie wartością aktualnie przybliżającą φ po wykonaniu k-tej iteracji.

Dla ustalenia uwagi niech $x > 0$.

- Zauważmy, iż jeśli $y_k > 0$, to następny przesuw γ_k należy odjąć od β_k.
- Jeśli $y_k < 0$, to następny przesuw γ_k należy dodać do β_k

Jeśli $x < 0$, to

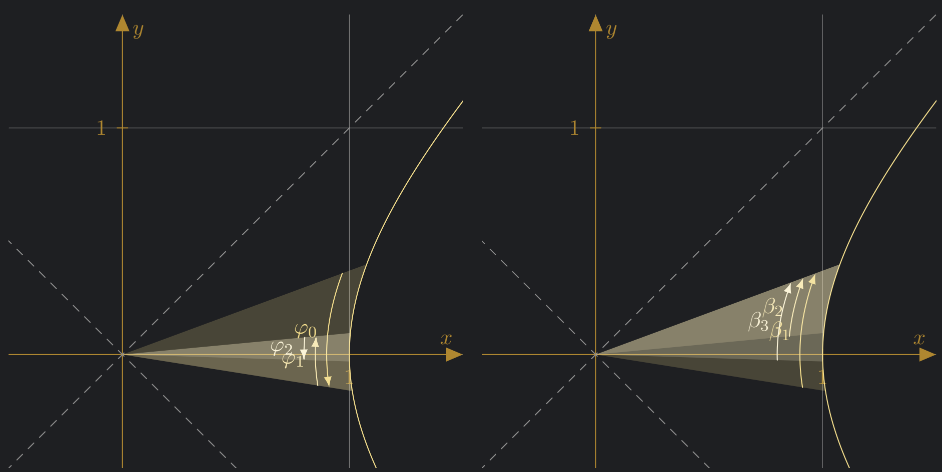
- Gdy $y_k < 0$ to wtedy następny γ_k należy odjąć
- Gdy $y_k > 0$ to wtedy następny γ_k należy dodać

Zbiorczo można zapisać $\varphi_k = -\operatorname{sgn}(x_k y_k)\gamma_k$ Suma takich przesunięć sprowadza punktu $A = (A_x, A_y)$ do punktu $A' = \left(\sqrt{A_x^2 - A_y^2}, 0\right)$

$$\sum_{k=0}^{n-1} \varphi_k = \sum_{k=0}^{n-1} - \operatorname{sgn}\left(x_k y_k\right) \gamma_k$$

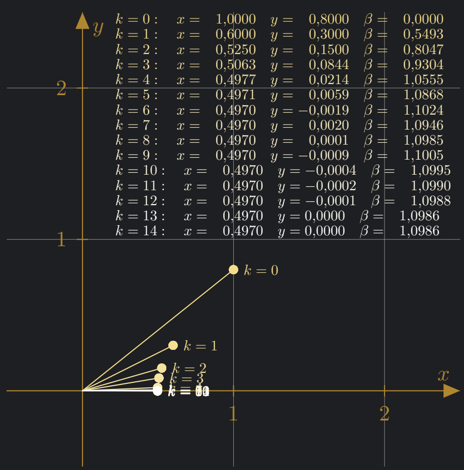
Weźmy jednak pod uwagę, iż β ma odwrotny zwrot do rozpatrywanej sumy. Prowadzi nas to do następującej rekurencji

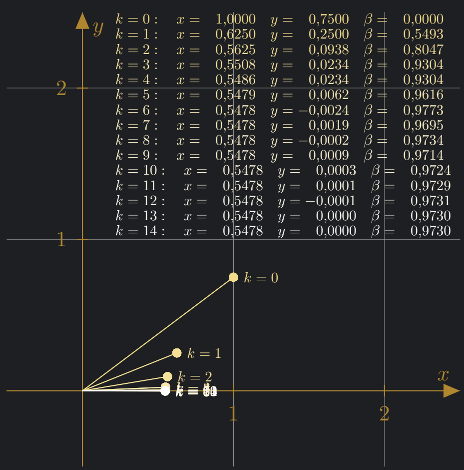
$$\begin{cases} \beta_{k+1} = \beta_k + \operatorname{sgn}\left(x_k y_k\right) \gamma_k \\ \beta_0 = 0 \end{cases}$$



$$
\begin{cases}
x_{k+1} = x_k + y_k \operatorname{tgh} \varphi_k \\
y_{k+1} = y_k + x_k \operatorname{tgh} \varphi_k \\
\beta_{k+1} = \beta_k - \varphi_k \\
x_0 = A_x \cdot \bar{H}_n \\
y_0 = A_y \cdot \bar{H}_n \\
\beta_0 = 0
\end{cases}
\rightarrow
\begin{cases}
\begin{cases}
\hat{x}_{j+1} = \hat{x}_j - \hat{y}_j \operatorname{sgn}(\hat{x}_j \hat{y}_j) 2^{-(k+1)} \\
\hat{y}_{j+1} = \hat{y}_j - \hat{x}_j \operatorname{sgn}(\hat{x}_j \hat{y}_j) 2^{-(k+1)} \\
\hat{\beta}_{j+1} = \hat{\beta}_j + \operatorname{sgn}(\hat{x}_j \hat{y}_j) \gamma_k \\
\hat{x}_0 = x_k \\
\hat{y}_0 = y_k \\
\hat{\beta}_0 = \beta_k
\end{cases} \\
t_k = \begin{cases} 2, & \text{dla } k+1 = 4, 13, 40, 121, \ldots, s, 3s+1, \ldots \\ 1, & \text{dla pozostałych} k+1 \end{cases} \\
x_{k+1} = \hat{x}_{t_k} \\
y_{k+1} = \hat{y}_{t_k} \\
\beta_{k+1} = \hat{\beta}_{t_k} \\
x_0 = A_x \cdot H_n \\
y_0 = A_y \cdot H_n \\
\beta_0 = 0
\end{cases}
$$

W tym trybie także możemy zapomnieć o stałych, ponadto po wykonaniu kilku iteracji możemy sobie wybrać najlepszą nie jesteśmy skazani na ostatnią.
Pierwszy przykład to obliczanie $\operatorname{artgh} \frac{4}{5}$. Drugi przykład to obliczanie $\operatorname{artgh} \frac{3}{4}$





1.8 Podsumowanie jako szybki praktyczny poradnik obsługi algorytmu CORDIC

Tryb	Rotacyjny	Wektorowy
Okrężny ($\mu = 1$)	$\sin\alpha$, $\cos\alpha$	$\operatorname{arctg} c$
Liniowy ($\mu = 0$)	$a \cdot b$	$c : a$
Hiperboliczny ($\mu = -1$)	$\sinh\varphi$, $\cosh\varphi$	$\operatorname{artgh} c$

$$\begin{cases} x_{k+1} = x_k - \mu d_k y_k 2^{-k} \\ y_{k+1} = y_k + d_k x_k 2^{-k} \\ \beta_{k+1} = \beta_k - d_k \gamma_k \\ x_0 = ? \\ y_0 = ? \\ \beta_0 = ? \end{cases}$$

- Rotacyjny $d_k = \operatorname{sgn}(\beta_k)$
- Wektorowy $d_k = -\operatorname{sgn}(x_k \cdot y_k)$
- Uwaga w trybie hiperbolicznym niektóre iteracje trzeba wykonać podwójnie

Na etapie projektowym należy stablicować wartości ciągów $(\gamma_k), (K_n), (H_n)$

- W trybie okrężnym

$$\gamma_k = \operatorname{arctg} 2^{-k} \quad \text{dla } k = 0, 1, 2, \ldots$$

$$K_n = \frac{1}{\sqrt{\prod_{k=0}^{n-1} (1 + 2^{-2k})}} \quad \text{dla } n = 1, 2, \ldots$$

- W Liniowy nie trzeba tablicować

$$\gamma_k = 2^{-k}, \text{ dla } k = 0, 1, 2, \ldots$$

- W Hiperboliczny

$$\gamma_k = \operatorname{artgh} 2^{-(k+1)} \quad \text{dla } k = 0, 1, 2, \ldots$$

$$H_n = \frac{1}{\sqrt{\prod_{k=0}^{n-1} \prod_{j=1}^{t_k} (1 - 2^{-2(k+1)})}} \quad \text{dla } n = 1, 2, \ldots$$

$$t_k = \begin{cases} 2, & \text{dla } k + 1 = 4, 13, 40, 121, \ldots, s, 3s+1, \ldots \\ 1, & \text{dla pozostałych } k + 1 \end{cases}$$

Co dokładniej znaczy powtórzyć niektóre iteracje w trybie hiperbolicznym?

$$\hat{d}_j = \operatorname{sgn}\left(\hat{\beta}_j\right)$$

$$\hat{d}_j = -\operatorname{sgn}(\hat{x}_j \hat{y}_j)$$

$$\begin{cases} \begin{cases} \hat{x}_{j+1} = \hat{x}_j - \mu \hat{d}_j \hat{y}_j 2^{-(k+1)} \\ \hat{y}_{j+1} = \hat{y}_j + \hat{d}_j \hat{x}_j 2^{-(k+1)} \\ \hat{\beta}_{j+1} = \hat{\beta}_j - \hat{d}_j \gamma_k \\ \hat{x}_0 = x_k \\ \hat{y}_0 = y_k \\ \hat{\beta}_0 = \beta_k \end{cases} \\ t_k = \begin{cases} 2, & \text{dla } k + 1 = 4, 13, 40, 121, \ldots, s, 3s+1, \ldots \\ 1, & \text{dla pozostałych } k + 1 \end{cases} \\ x_{k+1} = \hat{x}_{t_k} \\ y_{k+1} = \hat{y}_{t_k} \\ \beta_{k+1} = \hat{\beta}_{t_k} \end{cases}$$

1.8.1 Aby policzyć $\sin\varphi$ oraz $\cos\varphi$, gdzie $\varphi \in (-90°, 90°)$

1. Wybieramy n np. $n = 12$ i odczytujemy z tabelki wartość K_n
2. $x_0 = K_n \quad y_0 = 0 \quad \beta_0 = \varphi$
3. Wykonujemy n iteracji algorytmu
4. Odczytujemy $\sin\varphi \approx y_n$ oraz $\cos\varphi \approx x_n$

Co jak kąt jest z poza zakresu?

- Przy obliczaniu $\sin\alpha$ oraz $\cos\alpha$, dla dowolnego α
- Obliczamy $\tilde{\alpha} = \mod(\alpha + 90°, 360°) - 90°$
  - Jeżeli $\tilde{\alpha} > 90°$, to
    - $\varphi = \tilde{\alpha} - 180°$
    - Algorytmem obliczamy $\sin\varphi$ oraz $\cos\varphi$
    - W otrzymanych wynikach trzeba zmienić znak $\sin\alpha = -\sin\varphi \qquad \cos\alpha = -\cos\varphi$
  - W przeciwnym razie
    - $\varphi = \tilde{\alpha}$
    - Algorytmem obliczamy $\sin\varphi$ oraz $\cos\varphi$
    - W otrzymanych wynikach nie zmieniamy znaku $\sin\alpha = \sin\varphi \qquad \cos\alpha = \cos\varphi$
- Równie dobrze rachunki mogą być prowadzone w radianach, wykonujemy je analogicznie

1.8.2 Aby policzyć $\operatorname{arctg} c$, gdy $c \in \mathbb{R}$

- $x_0 = 1 \quad y_0 = c \quad \beta_0 = 0$
- Wykonujemy dowolną liczbę iteracji algorytmu
- Wybieramy tę iterację, w której y_k jest najbliżej 0
- Odczytujemy $\operatorname{arctg} c \approx \beta_k$

1.8.3 Aby policzyć $a \cdot b$, gdy $a \in \mathbb{R}$ oraz $b \in (-2, 2)$

- $x_0 = a \quad y_0 = 0 \quad \beta_0 = b$
- Wykonujemy dowolną liczbę iteracji algorytmu
- Wybieramy tę iterację, w której β_k jest najbliżej 0
- Odczytujemy $a \cdot b \approx y_k$

1.8.4 Aby policzyć $c : a$, gdy $a, c \in \mathbb{R}$ oraz $\frac{c}{a} \in (-2, 2)$

- $x_0 = a \quad y_0 = c \quad \beta_0 = 0$
- Wykonujemy dowolną liczbę iteracji algorytmu
- Wybieramy tę iterację, w której y_k jest najbliżej 0
- Odczytujemy $c : a \approx \beta_k$

1.8.5 Aby policzyć $\sinh\varphi$ oraz $\cosh\varphi$, gdy $\varphi \in (-1,11; 1,11)$

- Wybieramy n np. $n = 12$ i odczytujemy z tabelki wartość H_n
- $x_0 = H_n \quad y_0 = 0 \quad \beta_0 = \varphi$
- Wykonujemy n iteracji algorytmu
- Odczytujemy $\sinh\varphi \approx y_n$ oraz $\cosh\varphi \approx x_n$

1.8.6 Aby policzyć $\operatorname{artgh} c$, gdy $c \in (-1; 1)$

- $x_0 = 1 \quad y_0 = c \quad \beta_0 = 0$
- Wykonujemy dowolną liczbę iteracji algorytmu
- Wybieramy tę iterację, w której y_k jest najbliżej 0
- Odczytujemy $\operatorname{artgh} c \approx \beta_k$

1.9 Tryb hiperboliczny - przeindeksowany

Równie dobrze tryby hiperboliczne można opisać zaindeksować nieco inaczej. Oczywiście jeden i drugi opis jest równoznaczny.

$$\begin{aligned}
\gamma_k &= \operatorname{artgh} 2^{-k} \quad \text{dla } k = 1, 2, \dots \\
t_k &= \begin{cases} 2 & \text{dla } k = 4, 13, 40, 121, \dots, s, 3s+1, \dots \\ 1 & \text{dla pozostałych } k \end{cases} \\
H_n &= \frac{1}{\sqrt{\prod_{k=1}^{n} \prod_{j=1}^{t_k} (1 - 2^{-2k})}} \quad \text{dla } n = 1, 2, \dots
\end{aligned}$$

Rotacyjny	Wektorowy artgh
$\hat{d}_j = \operatorname{sgn}(\hat{\beta}_j)$	$\hat{d}_j = -\operatorname{sgn}(\hat{x}_j \hat{y}_j)$
$x_1 = H_n$	$x_1 = 1$
$y_1 = 0$	$y_1 = A_y$
$\beta_1 = \varphi$	$\beta_1 = 0$
$\cosh\varphi \approx x_{n+1}$	$\operatorname{artgh} c \approx \beta_{n+1}$
$\sinh\varphi \approx y_{n+1}$	

- $n \in \mathbb{N}^+$ - Liczba iteracji, podwójne liczę pojedynczo
- $k \in \{1, \dots, n\}$
- $\mu = -1$

$$\begin{cases}
\begin{cases}
\hat{x}_{j+1} = \hat{x}_j - \mu \hat{d}_j \hat{y}_j 2^{-k} \\
\hat{y}_{j+1} = \hat{y}_j + \hat{d}_j \hat{x}_j 2^{-k} \\
\hat{\beta}_{j+1} = \hat{\beta}_j - \hat{d}_j \gamma_k \\
\hat{x}_0 = x_k \\
\hat{y}_0 = y_k \\
\hat{\beta}_0 = \beta_k
\end{cases} \\
x_{k+1} = \hat{x}_{t_k} \\
y_{k+1} = \hat{y}_{t_k} \\
\beta_{k+1} = \hat{\beta}_{t_k}
\end{cases}$$