JOURNAL OF NUMERICAL ANALYSIS AND APPROXIMATION THEORY

J. Numer. Anal. Approx. Theory, vol. 54 (2025) no. 2, pp. 345–356, doi.org/10.33993/jnaat542-1640

ictp.acad.ro/jnaat

POWER SERIES FOR THE HALF WIDTH OF THE VOIGT FUNCTION, REDERIVED

JOACHIM WUTTKE

Abstract. The Voigt function is the convolution of a Gaussian and a Lorentzian. We rederive power series for its half width at half maximum for the limiting cases of near-Gaussian and near-Lorentzian line shapes. We thereby provide independent verification and slight corrections of the expansion coefficients reported by Wang et al (2022). Results are used in our implementation of function `voigt_hwhm` in the open-source library *libcerf*.

MSC. 33C15, 41A58, 33-04.

Keywords. Voigt function, Faddeeva function, Maclaurin series, asymptotic series, high-precision computation.

1. INTRODUCTION

The Voigt [8] function is the convolution of a Gaussian and a Lorentzian [5, 7.19.1]. It is frequently used to describe peak shapes in spectroscopy and diffraction. In such applications, it is customary to characterize peaks by their full or half width at half maximum (fwhm, hwhm).

Much work has been devoted to numerical approximations of the Voigt function and its half width. For most applications, considering experimental uncertainties, quite simple approximations would do. Nonetheless, for internal consistency and reproducibility of data analyses, to facilitate numerical differentiation (e.g., in fitting routines), and out of pure mathematical interest, it is worthwhile to push approximations to ever better accuracy.

The Voigt function can be expressed through the Faddeeva function, which is closely related to the complex error function. Therefore, our open-source complex error function library *libcerf* [4] also provides functions `voigt` and `voigt_hwhm`. Earlier versions of *libcerf* computed `voigt_hwhm` very directly by bisection on function values of `voigt`. Starting with release 3.3, a completely new implementation of `voigt_hwhm` uses series expansions for the near-Gaussian and near-Lorentzian limiting cases, and piecewise Chebyshev interpolation [10] in between. Near machine accuracy is achieved in each region. This paper documents the two series expansions.

Forschungszentrum Jülich GmbH, JCNS at MLZ, Garching, Germany
j.wuttke@fz-juelich.de, ORCID 0000-0002-4028-1447.

The same series have recently been studied by Wang et al [9]. Here, we rederive them in different ways and thereby provide independent verification and slight corrections for the coefficients tabulated in [9]. For the near-Gaussian case, our derivation (Section 3) is more compact, as we use complex notation and represent the Voigt function through the Faddeeva function instead of decomposing it as in [9, Eq 9]. For the near-Lorentzian case, Wang et al chose a pragmatic method: numerical differentiation, followed by rounding to integer numerators [9, Appendix B]. Here (Section 4), we derive a fully algebraic recursion.

2. FOUNDATIONS

2.1. Voigt function. The Voigt function

$$V(q;\sigma,\gamma) := \int_{-\infty}^{\infty} \mathrm{d}p\, G(p;\sigma) L(q-p;\gamma) \tag{1}$$

is the convolution of a normalized Gaussian with standard deviation σ

$$G(q;\sigma) := \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{q^2}{2\sigma^2}\right) \tag{2}$$

and a normalized Lorentzian with half width γ

$$L(q;\gamma) := \frac{\gamma}{\pi\left(\gamma^2+q^2\right)}. \tag{3}$$

As G and L are even functions of σ and γ, so is V. Therefore in the following we will only consider $\sigma, \gamma \geq 0$. Otherwise said, we will simply write σ and γ instead of their absolute values.

2.2. Half width. Function $H(\sigma,\gamma)$ is the half width at half maximum of the Voigt function. It is implicitly defined by

$$V(H(\sigma,\gamma);\sigma,\gamma) = \tfrac{1}{2} V(0;\sigma,\gamma). \tag{4}$$

Or, introduce the reduced Voigt function

$$v(q;\sigma,\gamma) := \frac{V(q;\sigma,\gamma)}{V(0;\sigma,\gamma)}, \tag{5}$$

and its inverse function v^{-1} with respect to the first argument, such that

$$v(q;\sigma,\gamma) = y \;\Leftrightarrow\; v^{-1}(y;\sigma,\gamma) = q. \tag{6}$$

Then we have simply

$$H(\sigma,\gamma) = v^{-1}\left(\tfrac{1}{2};\sigma,\gamma\right). \tag{7}$$

2.3. Scaling. For real $\alpha \neq 0$, the Voigt function scales as

$$V(\alpha q; \alpha\sigma, \alpha\gamma) = \tfrac{1}{\alpha} V(q; \sigma, \gamma). \tag{8}$$

The reduced Voigt function scales accordingly as

$$v(\alpha q; \alpha\sigma, \alpha\gamma) = v(q; \sigma, \gamma), \tag{9}$$

and its inverse as

$$v^{-1}(y; \alpha\sigma, \alpha\gamma) = \alpha v^{-1}(y; \sigma, \gamma). \tag{10}$$

This implies for the half width

$$H(\alpha\sigma, \alpha\gamma) = \alpha H(\sigma, \gamma). \tag{11}$$

The homogeneity of degree 1 implies that we only need to study a function of one real variable σ/γ or γ/σ.

2.4. Relation to the Faddeeva function. Given the error function

$$\operatorname{erf}(z) := \frac{2}{\sqrt{\pi}} \int_0^z \mathrm{d}t\, \mathrm{e}^{-t^2}, \tag{12}$$

the complementary error function

$$\operatorname{erfc}(z) := 1 - \operatorname{erf}(z) \tag{13}$$

$$= \frac{2}{\sqrt{\pi}} \int_z^\infty \mathrm{d}t\, \mathrm{e}^{-t^2}, \tag{14}$$

and the underflow-compensated complementary error function

$$\operatorname{erfcx}(z) := \mathrm{e}^{z^2} \operatorname{erfc}(z), \tag{15}$$

the Faddeeva function is [5, 7.2.3]

$$w(z) = \operatorname{erfcx}(-iz) \tag{16}$$

$$= e^{-z^2} \operatorname{erfc}(-iz) \tag{17}$$

$$= \mathrm{e}^{-z^2} \frac{2}{\sqrt{\pi}} \int_{-iz}^\infty \mathrm{d}t\, \mathrm{e}^{-t^2} \tag{18}$$

$$= \mathrm{e}^{-z^2} \left(1 - \frac{2}{\sqrt{\pi}} \int_0^{-iz} \mathrm{d}t\, \mathrm{e}^{-t^2}\right) \tag{19}$$

$$= \mathrm{e}^{-z^2} \left(1 + i\frac{2}{\sqrt{\pi}} \int_0^{z} \mathrm{d}t\, \mathrm{e}^{t^2}\right). \tag{20}$$

Another integral representation, valid for $\operatorname{Im} z > 0$, is [5, 7.7.2]

$$w(z) = \frac{i}{\pi} \int_{-\infty}^{+\infty} \mathrm{d}t\, \frac{\mathrm{e}^{-t^2}}{z - t}. \tag{21}$$

Let us define $u := \operatorname{Re} w$ and $v := \operatorname{Im} w$ so that

$$w(z) \equiv u(z) + iv(z). \tag{22}$$

The real part of (21) is

$$u(x+iy) = \frac{y}{\pi}\int_{-\infty}^{+\infty} \mathrm{d}t \, \frac{\mathrm{e}^{-t^2}}{(x-t)^2+y^2}, \tag{23}$$

which is essentially Voigt's convolution integral (1). Divide t, x, and y by $\sqrt{2}\sigma$ to find

$$V(q;\sigma,\gamma) = \frac{1}{\sqrt{2\pi}\sigma} u\left(\frac{q}{\sqrt{2}\sigma} + i\frac{\gamma}{\sqrt{2}\sigma}\right). \tag{24}$$

Accordingly, the Voigt half width is implicitly defined by a functional equation that involves the real part of Faddeeva's function,

$$u\left(\frac{H(\sigma,\gamma)}{\sqrt{2}\sigma} + i\frac{\gamma}{\sqrt{2}\sigma}\right) = \frac{1}{2} u\left(i\frac{\gamma}{\sqrt{2}\sigma}\right). \tag{25}$$

3. NEAR-GAUSSIAN CASE $\gamma \ll \sigma$

3.1. Case $\gamma = 0$. In the case $\gamma = 0$, the Voigt function is just a Gaussian with standard deviation σ. With a real argument $z = x$, the real part of (20) is simply

$$u(x) = \mathrm{e}^{-x^2}. \tag{26}$$

The implicit equation (25) is solved by the well-known

$$H(\sigma,0) = \sqrt{2\ln 2}\,\sigma. \tag{27}$$

3.2. Rescaled function f. We now consider the case of a relatively small Lorentzian width, $\gamma \ll \sigma$. Thanks to the scaling property (11), the half width of the Voigt function can be computed as

$$H(\sigma,\gamma) = \sigma H(1,\gamma/\sigma). \tag{28}$$

For later convenience, we introduce the reduced variable

$$t := \frac{\gamma}{\sqrt{2}\sigma} \tag{29}$$

so that

$$H(\sigma,\gamma) = \sigma H(1,\sqrt{2}t). \tag{30}$$

We define the function

$$f(t) := \frac{H(1,\sqrt{2}t) - H(1,0)}{\sqrt{2}} + it \tag{31}$$

and the constant

$$\eta := \frac{H(1,0)}{\sqrt{2}} = \sqrt{\ln 2} \tag{32}$$

so that $H(\sigma,\gamma)$ can be obtained via (30) and

$$H(1,\sqrt{2}t) = \sqrt{2}\,(\eta + f(t) - it)\,. \tag{33}$$

Below, we will determine a series expansion for $f(t)$, so that we can ultimately compute the Voigt width using

$$H(1, \sqrt{2}t) = \sqrt{2}\left(\eta + \sum_{n=1}^{\infty} (\operatorname{Re} f_n) t^n\right). \tag{34}$$

3.3. Power series in t and f. In (25), choose $\sigma = 1$ and $\gamma = \sqrt{2}t$. then use (30) and (33) to obtain an implicit equation for f,

$$u\left(\eta + f(t)\right) = \tfrac{1}{2} u\left(it\right). \tag{35}$$

On the right-hand side, use the Maclaurin expansion of $w(z)$ [5, 7.6.3] to obtain

$$u(it) = \sum_{n=0}^{\infty} \frac{(-t)^n}{\Gamma(\frac{n}{2}+1)}. \tag{36}$$

On the left-hand side of (35), use the Taylor expansion

$$u\left(\eta + f(t)\right) = \operatorname{Re} \sum_{n=0}^{\infty} w_n f(t)^n \tag{37}$$

with coefficients

$$w_n := \frac{w^{(n)}(\eta)}{n!} \tag{38}$$

given by the recursion [5, 7.10.3] [7, Eq 10]

$$w_n = -\frac{2}{n}\left(\eta w_{n-1} + w_{n-2}\right) \tag{39}$$

with starting values $w_0 := w(\eta)$ and $w_{-1} := -i/\sqrt{\pi}$.

Combine (35–37), and omit terms in t^0, to obtain an equation that relates the power series in t and in f,

$$\tfrac{1}{2} \sum_{n=1}^{\infty} \frac{(-t)^n}{\Gamma(\frac{n}{2}+1)} = \sum_{n=1}^{\infty} \operatorname{Re} w_n f(t)^n. \tag{40}$$

3.4. Recursive solution. To solve (40) numerically, function $f(t)$ shall be expressed as a power series in t. Since $f(0) = 0$ by construction (31), the power series starts with the linear term,

$$f(t) \equiv \sum_{n=1}^{\infty} f_n t^n. \tag{41}$$

Per (31), the f_n are real for $n > 1$; only for $n = 1$ there is an imaginary part, per the definition (31), so that

$$\operatorname{Im} f_n = \delta_{n1}. \tag{42}$$

With notation from [3], write $[t^N]f(t)$ for the coefficient of t^N in the power series representing $f(t)$. For the series (41), $[t^N]f(t) = f_N$. Define

$$F_{N,n} := \left[t^N\right] f(t)^n, \tag{43}$$

and apply the operator $[t^N]$ to both sides of (40):

$$\frac{(-1)^N}{2\Gamma(\frac{N}{2}+1)} = \sum_{n=1}^{N} \operatorname{Re} w_n F_{N,n} \tag{44}$$

for all $N \geq 1$.

Equations (41, 43, 44) fully determine the f_n in terms of the w_n, which are given by the recursion (39). We resolve said equations in two steps. First, we solve (44) for $F_{N,1}$. For $N = 1$, we recall (42) to obtain

$$F_{1,1} = (\operatorname{Re} w_1)^{-1} \left(\operatorname{Im} w_1 - \frac{1}{2\Gamma(\frac{3}{2})} \right) + i. \tag{45}$$

For $N > 1$,

$$F_{N,1} = (\operatorname{Re} w_1)^{-1} \left(\frac{(-1)^N}{2\Gamma(\frac{N}{2}+1)} - \sum_{n=2}^{N} \operatorname{Re} w_n F_{N,n} \right). \tag{46}$$

Second, we use (41) to obtain the $F_{N,n}$ for $n > 1$:

$$F_{N,n} = \sum_{m=1}^{N-n+1} F_{m,1} F_{N-m,n-1}. \tag{47}$$

Finally, we note that the coefficients of (41) are $f_N \equiv F_{N,1}$. This completes our derivation of a power series for the half width H.

To minimize the number of computational operations in the highly optimized code of *libcerf*, we write (28, 34) as

$$H(\sigma,\gamma) = \sigma\sqrt{2\ln 2} + \gamma \sum_{n=0}^{N_{\text{cut}}} c_n \left(\frac{\gamma}{\sigma}\right)^n \tag{48}$$

with precomputed coefficients

$$c_n := \frac{\operatorname{Re} f_{n+1}}{\sqrt{2}^n}. \tag{49}$$

3.5. Implementation. A Python script `near_gauss.py` used to generate coefficients is preserved in subdirectory `dev/voigt` of the *libcerf* source repository. The script uses the arbitrary-precision math library *mpmath* [6] with a fixed internal precision, checked against a computation with some more digits. It turns out that an internal precision of 40 decimal digits is needed to compute f_n for $n < 32$ with 32 reliable decimal digits. The program executes within a fraction of a second.

The recursive forward computation of w_n according to (39) would be divergent if η had a positive imaginary part [7]. In our case, η is real so that we are at the margin of numerical instability. Indeed, our computations show that up to $n = 30$ only the last few decimal digits get incorrect. Nonetheless, our script validates each w_n against a non-recursive computation, derived in the Appendix.

f_1	$=$	$8.3255461115769775635316464489 52 \cdot 10^{-1} + i$
f_2	$=$	$5.3254711842961210323020845059416 \cdot 10^{-1}$
f_3	$=$	$1.3603423870145348659601346974136 \cdot 10^{-1}$
f_4	$=$	$-6.3839925995348583105863651935208 \cdot 10^{-3}$
f_5	$=$	$-7.5882994178697868047017954181619 \cdot 10^{-3}$
f_6	$=$	$7.5685451134845100193553849814044 \cdot 10^{-4}$
f_7	$=$	$6.4174309726033170181322853645455 \cdot 10^{-4}$
f_8	$=$	$-1.0278614365257442345642575963235 \cdot 10^{-5}$
f_9	$=$	$-6.6864392638387619203117167133824 \cdot 10^{-5}$
f_{10}	$=$	$-1.8800729899141457354675112660009 \cdot 10^{-5}$
f_{11}	$=$	$9.3901358253570724565409358708571 \cdot 10^{-6}$
f_{12}	$=$	$5.4149990265667553408636905696295 \cdot 10^{-6}$
f_{13}	$=$	$-1.2862976252461744893956942201673 \cdot 10^{-6}$
f_{14}	$=$	$-1.0759168918380548822306060203341 \cdot 10^{-6}$
f_{15}	$=$	$7.8733635964790862989086501951507 \cdot 10^{-8}$
f_{16}	$=$	$1.9255725519174188542320412973488 \cdot 10^{-7}$
f_{17}	$=$	$2.5308903977393059634088084205148 \cdot 10^{-8}$
f_{18}	$=$	$-3.3104307709547517055285672959576 \cdot 10^{-8}$
f_{19}	$=$	$-1.1821070040002130133075915099552 \cdot 10^{-8}$
f_{20}	$=$	$5.0020607880755762331999675884955 \cdot 10^{-9}$
f_{21}	$=$	$3.2040951850692659104678394048668 \cdot 10^{-9}$
f_{22}	$=$	$-4.9276721508012916216290609360574 \cdot 10^{-10}$
f_{23}	$=$	$-7.1352246104725448681836423474852 \cdot 10^{-10}$
f_{24}	$=$	$-3.24079995213\underline{73985008284717584509} \cdot 10^{-11}$
f_{25}	$=$	$1.401088301440\underline{9870068385079612875} \cdot 10^{-10}$
f_{26}	$=$	$3.3772678382\underline{910580581437551000254} \cdot 10^{-11}$
f_{27}	$=$	$-2.3680267709\underline{337763087056901649239} \cdot 10^{-11}$
f_{28}	$=$	$-1.14626868307\underline{07711798806492528473} \cdot 10^{-11}$
f_{29}	$=$	$3.003967044\underline{3183376826912340278726} \cdot 10^{-12}$
f_{30}	$=$	$2.94788896\underline{14035047804083367602927} \cdot 10^{-12}$
f_{31}	$=$	$-9.746764\underline{6700490397869815620681195} \cdot 10^{-14}$

Table 1. Coefficients f_n of the series (41) that can be used according to (33) for computing the half width $H(\sigma, \gamma)$ in the near-Gaussian case of $\gamma \ll \sigma$. Digits that deviate from [9, Table A1] are underlined. Agreement through $n = 22$ confirms essential correctness of both their work and ours.

3.6. Results. Resulting coefficients f_n up to $n = 30$ are reported in Table 1 with 32 digits precision. Up to $n = 22$, they fully agree with [9, Table A1]. Then, suddenly, the agreement drops to 12 digits for f_{23}, and further decreases to 7 digits for f_{30}. Given that our algorithm is simpler than the one of [9] and that we cross-checked our results with different internal accuracies we are confident that all digits reported in Table 1 are correct.

4. NEAR-LORENTZIAN CASE $\sigma \ll \gamma$

4.1. Case $\sigma = 0$. For a pure Lorentzian, $V(q;0,\gamma) = L(q;\gamma)$, the half width is just $H(0,\gamma) = \gamma$.

4.2. Rescaled function g. We now consider the case of a relatively small Gaussian width, $\sigma \ll \gamma$. Thanks to the scaling property (11), the half width of the Voigt function can be computed as

$$H(\sigma,\gamma) = \gamma H(\sigma/\gamma, 1). \tag{50}$$

For later convenience, we introduce a reduced variable

$$s := \sigma^2/\gamma^2 \tag{51}$$

and a real function

$$g(s) := H\left(\sqrt{s}, 1\right) - 1 = H\left(\sigma/\gamma, 1\right) - 1 = \frac{H(\sigma,\gamma) - \gamma}{\gamma} \tag{52}$$

that is to be determined so that we ultimately obtain the Voigt half width as

$$H(\sigma,\gamma) = \gamma\left(1 + g\left(\sigma^2/\gamma^2\right)\right). \tag{53}$$

In this notation, the implicit definition (25) of H can be written as

$$2u\left(\frac{1+g(s)+i}{\sqrt{2s}}\right) - u\left(\frac{i}{\sqrt{2s}}\right) = 0, \tag{54}$$

which needs to be resolved for g.

For large $|z|$, the Faddeeva function can be computed from its asymptotic expansion [1, p. 14]

$$w(z) = \frac{i}{\sqrt{\pi}} \sum_{n=0}^{N} \frac{a_n}{2^n z^{2n+1}} + R_N(z) \tag{55}$$

with the abbreviation

$$a_n := \frac{2^n}{\sqrt{\pi}} \Gamma\left(n + 1/2\right) = \frac{(2n)!}{2^n n!} = (2n-1)!!. \tag{56}$$

The remainder $|R_N|$ is no larger than the first term omitted from the sum. Drop the explicit notation of truncation, and apply (55) to (54),

$$0 \simeq \operatorname{Im} \sum_{n=0}^{\infty} \frac{a_n}{2^n} \left\{ 2\left(\frac{\sqrt{2s}}{1+g+i}\right)^{2n+1} - \left(\frac{\sqrt{2s}}{i}\right)^{2n+1} \right\}. \tag{57}$$

Factor out $(2s)^n$, and drop a common factor $\sqrt{2s}$:

$$0 \simeq \operatorname{Im} \sum_{n=0}^{\infty} a_n \left\{ 2\left(\frac{1}{1+g+i}\right)^{2n+1} - \left(\frac{1}{i}\right)^{2n+1} \right\} s^n. \tag{58}$$

Binomial expansion:

$$0 \simeq \operatorname{Im} \sum_{n=0}^{\infty} a_n \left\{ 2 \left(\frac{1}{1+i} \right)^{2n+1} \sum_{k=0}^{\infty} \binom{2n+k}{k} \left(\frac{-g}{1+i} \right)^k - \left(\frac{1}{i} \right)^{2n+1} \right\} s^n. \tag{59}$$

As a_n, s, g are all real, the only complex quantity here is the imaginary unit i.

We precompute

$$(-1)^k \operatorname{Im} \left(\frac{1}{1+i} \right)^{2n+k+1} = \frac{\psi_{2n+k}}{4 \cdot 4^{\lfloor (2n+k)/4 \rfloor}}, \tag{60}$$

where ψ_k is an integer function with period 8, given by

$$(\psi_0, \psi_1, \ldots, \psi_7) \coloneqq (2, -2, 1, 0, -2, 2, -1, 0). \tag{61}$$

Multiply (59) by 2 and use (60) to obtain

$$0 \simeq \sum_{n=0}^{\infty} a_n \left\{ \sum_{k=0}^{\infty} b_{n,k} g^k - 2(-1)^n \right\} s^n. \tag{62}$$

with the rational function

$$b_{n,k} \coloneqq \binom{2n+k}{k} \frac{\psi_{2n+k}}{4^{\lfloor (2n+k)/4 \rfloor}}. \tag{63}$$

4.3. Solution. We seek a solution in form of a power series

$$g(s) = \sum_{j=1}^{\infty} g_j s^j. \tag{64}$$

As in Section 3.4, write $[s^n]f$ for the coefficient of s^n in a power series representing $f(s)$. Apply the operator $[s^n]$ to (62):

$$0 \simeq \sum_{m=0}^{\infty} a_m \left\{ (b_{m,0} - 2(-1)^m) + [s^{n-m}] \sum_{k=1}^{\infty} b_{m,k} g^k \right\}. \tag{65}$$

Define

$$G_{N,m} \coloneqq \left[s^N \right] g^m. \tag{66}$$

Restrict the sums to non-zero terms, using the fact that $g(s)$ per (64) has no s^0 term:

$$0 \simeq a_n (b_{n,0} - 2(-1)^n) + \sum_{m=0}^{n} a_m \sum_{k=1}^{n-m} b_{m,k} G_{n-m,k}. \tag{67}$$

The only linear occurrence of $g_n \equiv G_{n,1}$ is in the $m = 0, k = 1$ term, with prefactor $a_0 b_{0,1} = -2$. Isolating that contribution gives the recurrence

$$G_{n,1} = \frac{1}{2} \left\{ a_n (b_{n,0} - 2(-1)^n) + \sum_{m=0}^{n} a_m \sum_{k=1+\delta_{m0}}^{n-m} b_{m,k}\, G_{n-m,k} \right\}. \tag{68}$$

$g_0 =$	$1/2^0$	$=$	1.	
$g_1 =$	$3/2^1$	$=$	1.5	
$g_2 =$	$-21/2^3$	$=$	-2.625	
$g_3 =$	$183/2^4$	$=$	11.4375	
$g_4 =$	$-10413/2^7$	$=$	-81.3515625	
$g_5 =$	$198477/2^8$	$=$	775.30078125	
$g_6 =$	$-9070497/2^{10}$	$=$	-8857.9072265625	
$g_7 =$	$241045983/2^{11}$	$=$	$117698.2338867187\ldots$	
$g_8 =$	$-58945112829/2^{15}$	$=$	$-1798862.085845947\ldots$	
$g_9 =$	$2038148025489/2^{16}$	$=$	$31099670.79908752\ldots$	
$g_{10} =$	$-156915708321627/2^{18}$	$=$	$-598585923.4681205\ldots$	
$g_{11} =$	$6654666968153361/2^{19}$	$=$	$12692769943.52981\ldots$	
$g_{12} =$	$-1234122510349636713/2^{22}$	$=$	$-294237735354.8137\ldots$	
$g_{13} =$	$62113630324702350897/2^{23}$	$=$	$7404521742427.628\ldots$	
$g_{14} =$	$-6745007796528337910073/2^{25}$	$=$	$-201016896859655.9\ldots$	
$g_{15} =$	$393001502942654465122863/2^{26}$	$=$	$5856178744772888.\ldots$	

Table 2. Coefficients g_n of the asymptotic expansion (64) that can be used according to (53) for computing the half width $H(\sigma,\gamma)$ in the near-Lorentzian case of $\sigma \ll \gamma$. Based on these g_n, we can confirm the T_{2n} of [9, Table A2] except for $n = 15$, as discussed in Section 4.5.

The recurrence starts from

$$G_{1,1} = \frac{a_1(b_{1,0}+2)}{2} = \frac{3}{2}. \tag{69}$$

Coefficients $G_{n,k}$ with $1 < k \le n$ can be computed by another recursion, as in (47) of Section 3.4,

$$G_{n,k} = \sum_{j=1}^{n-k+1} G_{j,1} G_{n-j,k-1}. \tag{70}$$

While the coefficients a, c in (68) are integers, the b are rational numbers with denominators that are powers of 2. Therefore the $g_n \equiv G_{n,1}$ must also be rational numbers, and their denominators powers of 2.

4.4. **Implementation.** A Python script `near_lorentz.py` used to generate coefficients is preserved in subdirectory `dev/voigt` of the *libcerf* source repository. The script employs the Python library *Fraction* to compute exact rational numbers using long-integer arithmetics.

4.5. **Results.** The expansion coefficients g_n are reported in Table 2. The ratio of consecutive coefficients is shown as $\log|g_n/g_{n-1}|$ versus n in Figure 1, which indicates that the $|g_n|$ grow faster than a power law. In consequence, (64) is an asymptotic series, as was noted by Wang et al [9], and was to be expected from the asymptotic expansion (55) used for $w(z)$.

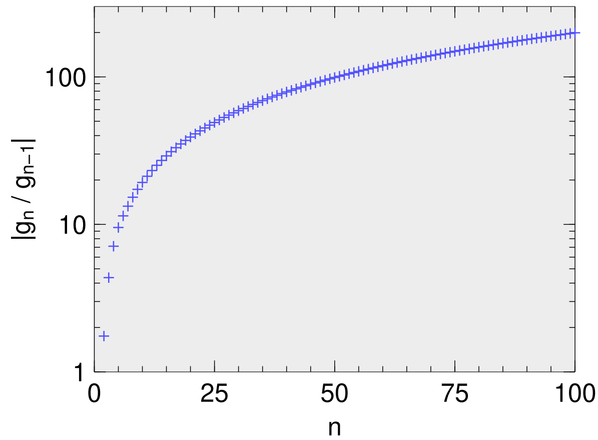

Fig. 1. Absolute value of the ratio of consecutive coefficients, $|g_n/g_{n-1}|$, shown on a logarithmic scale against n. The ratio appears to grow slowly beyond all limits.

Wang et al expanded squared half widths in a series [9, Eq A30] that reads in our notation

$$\left(\frac{H(\sigma,\gamma)}{\gamma}\right)^2 = \sum_{n=0}^{\infty} T_n \left(\frac{\sigma}{\sqrt{2}\gamma}\right) = \sum_{n=0}^{\infty} T_n \left(\sqrt{\frac{s}{2}}\right). \tag{71}$$

They computed rational numbers T_n by approximating floating-point numbers obtained from numerical differentiation [9, Eq A31]. For odd n, they found $T_n = 0$, consistent with our (53, 64). To validate their results for even n, we recompute T_{2m} as

$$T_{2m} = \left[\left(\frac{s}{2}\right)^m\right] \left(\frac{H(\sigma,\gamma)}{\gamma}\right)^2 = 2^m \,[s^m]\,(1+g(s))^2. \tag{72}$$

We define $g_0 := 1$ so that we are left with a self-convolution

$$T_{2m} = 2^m \,[s^m] \left(\sum_{k=0}^{\infty} g_k s^k\right)^2 = 2^m \sum_{k=0}^{m} g_k g_{m-k}. \tag{73}$$

With rational arithmetics based on our g_k as shown in Table 2, we confirm the T_{2m} of [9, Table A2] except for their $T_{30} = 176955754371862947/2^{19}$, which is too large by $3/2^{20}$; the correct result is $353911508743725891/2^{20}$.

APPENDIX. CLOSED EXPRESSION FOR w_n

As the recursion (39) is marginally unstable, it may be preferable to compute w_n from a closed expression, or at least use the direct computation as a cross-check. With (17, 38),

$$w_n = \frac{1}{n!}\frac{\mathrm{d}^n}{\mathrm{d}(-iz)^2} \mathrm{e}^{(-iz)^2} \operatorname{erfc}(iz). \tag{74}$$

With [5, 7.18.4], [2, 2.23], and with the notation $\mathrm{i}^n \operatorname{erfc}$ [5, 7.18.2] for the iterated integral of erfc,

$$w_n = i^n 2^n \mathrm{e}^{(-iz)^2} \mathrm{i}^n \operatorname{erfc}(-iz). \tag{75}$$

With [5, 7.18.11],

$$w_n = i^n 2^n \frac{\sqrt{2}^{n-1}}{\sqrt{\pi}} \mathrm{e}^{z^2/2} U\big(n + 1/2, -i\sqrt{2}z\big). \tag{76}$$

The parabolic cylinder function U [5, Chapter 12] is provided by *mpmath* as the function `pcfu`. Internally, *mpmath* computes it by means of hypergeometric functions.

REFERENCES

[1] V.N. Faddeyeva, N.M. Terent'ev, *Tables of Values of the Function* $w(z) = \mathrm{e}^{-z^2}\Big(1 + \frac{2i}{\sqrt{\pi}}\int_0^z \mathrm{e}^{t^2}\,\mathrm{d}t\Big)$ *for Complex Argument*, Pergamon, Oxford, 1961.

[2] W. Gautschi, *Efficient computation of the complex error function*, SIAM J. Numer. Anal. **7** (1970), pp. 187–198. `https://doi.org/10.1137/0707012`

[3] R.L. Graham, D.E. Knuth, O. Patashnik, *Concrete Mathematics*, 1989, Addison–Wesley, Reading.

[4] S.G. Johnson, J. Wuttke (2013–2025), *libcerf, a numeric library providing complex error functions*, `https://jugit.fz-juelich.de/mlz/libcerf`

[5] NIST Digital Library of Mathematical Functions. `http://dlmf.nist.gov`

[6] The mpmath development team (2023), *mpmath: a Python library for arbitrary-precision floating-point arithmetic (version 1.3.0)*, `https://mpmath.org`

[7] I. Thompson, *Algorithm 1046: An Improved Recurrence Method for the Scaled Complex Error Function*, ACM Trans. Math. Software, **50** (2024), no. 20 (18 pages). `https://doi.org/10.1145/3688799`

[8] W. Voigt, *Über das Gesetz der Intensitätsverteilung innerhalb der Linien eines Gasspektrums*, Sitzungsber. Bayer. Akad. Wiss. Math.-Naturwiss. Kl., **25** (1912), pp. 603–620. `https://publikationen.badw.de/de/003395768`

[9] Y. Wang, B. Zhou, R. Zhao, B. Wang, Q. Liu, M. Dai, *Super-Accuracy Calculation for the Half Width of a Voigt Profile*, Mathematics, **10** (2022), no. 210 (14 pages). `https://doi.org/10.3390/math10020210`

[10] J. Wuttke, A. Kleinsorge, *Algorithm 1xxx: Code generation for piecewise Chebyshev approximation*, submitted to ACM TOMS.

Received by the editors: November 05, 2025; accepted: November 21, 2025; published online: December 15, 2025.