Interpolation

Contents

1 Lagrange interpolation 1
1.1 Generalised interpolation 2
1.2 Bi-dimensional interpolation 4
1.3 Derivative and integral of an interpolated function 4

In this part of the documentation we address the question of interpolation. Interpolation is an extremely powerful tool that allows one to reconstruct, within some accuracy, continuos functions when knowing them only in a finite number of points. Unsurprisingly, interpolation techniques are massively used in numerical codes that deal with physical quantities that are continuos functions of space-time variables, such as position and momentum. In this respect, APFEL++ makes no exception.

In order to make the best of interpolation, APFEL++ has been designed around a specific choice of the interpolation strategy. Specifically, the very computational core of APFEL++ relies on *Langrange* polynomials and their properties. Despite Langrange polynomials do not enjoy the smoothness of splines, they enjoy a number of very useful properties that allow extending the use of Lagrange polynomials from just interpolation to derivation and integration. In the following, the interpolation strategy is detailed along with its implementation and applications in APFEL++.

1 Lagrange interpolation

In this section we will derive a general expression for the Lagrange interpolating functions w. These functions are ubiquitous in APFEL++ in that they are used, not only for simple interpolations, but also for computing convolutions, integrals in general, and also derivatives. It is then important to give them a thorough derivation in terms of Lagrange polynomials and to understand how they behave upon derivation and integration.

Suppose one wants to interpolate the test function g in the point x using a set of Lagrange polynomials of degree k. This requires a subset of $k+1$ consecutive points on an interpolation grid, say $\{x_\alpha, \ldots, x_{\alpha+k}\}$. The relative position between the point x and the subset of points used for the interpolation is arbitrary. It is convenient to choose the subset of points such that $x_\alpha < x \leq x_{\alpha+k}$.[1] However, the ambiguity remains because there are k possible choices according to whether $x_\alpha < x \leq x_{\alpha+1}$, or $x_{\alpha+1} < x \leq x_{\alpha+2}$, and so on. For now we assume that:

$$x_\alpha < x \leq x_{\alpha+1} \,, \tag{1.1}$$

but we will release this assumption below. Using the standard Lagrange interpolation procedure, one can approximate the function g in x as:

$$g(x) = \sum_{i=0}^{k} \ell_i^{(k)}(x) g(x_{\alpha+i}) \,, \tag{1.2}$$

where $\ell_i^{(k)}$ is the i-th Lagrange polynomial of degree k which can be written as:

$$\ell_i^{(k)}(x) = \prod_{\substack{m=0 \\ m\neq i}}^{k} \frac{x - x_{\alpha+m}}{x_{\alpha+i} - x_{\alpha+m}} \,. \tag{1.3}$$

[1] Actually, it is not necessary to impose the constraint $x_\alpha < x \leq x_{\alpha+k}$. In case this relation is not fulfilled one usually speaks about *extrapolation* rather than *interpolation*. If not necessary, this option is typically not convenient because it may lead to a substantial deterioration in the accuracy with which $g(x)$ is determined.

Since we have assumed that $x_\alpha < x \leq x_{\alpha+1}$ (see Eq. (1.1)), Eq. (1.2) can be written as:

$$g(x) = \theta(x - x_\alpha)\theta(x_{\alpha+1} - x) \sum_{i=0}^{k} g(x_{\alpha+i}) \prod_{\substack{m=0 \\ m\neq i}}^{k} \frac{x - x_{\alpha+m}}{x_{\alpha+i} - x_{\alpha+m}} \,. \tag{1.4}$$

In order to make Eq. (1.4) valid for all values of α, one just has to sum over all $N_x + 1$ nodes of the *global* interpolation grid $\{x_0, \ldots, x_{N_x}\}$, that is:

$$g(x) = \sum_{\alpha=0}^{N_x-1} \theta(x - x_\alpha)\theta(x_{\alpha+1} - x) \sum_{i=0}^{k} g(x_{\alpha+i}) \prod_{\substack{m=0 \\ m\neq i}}^{k} \frac{x - x_{\alpha+m}}{x_{\alpha+i} - x_{\alpha+m}} \,. \tag{1.5}$$

Defining $\beta = \alpha + i$, one can rearrange the equation above as:

$$g(x) = \sum_{\beta=0}^{N_x+k-1} w_\beta^{(k)}(x) g(x_\beta) \,, \tag{1.6}$$

that leads to the definition of the interpolating functions:

$$w_\beta^{(k)}(x) = \sum_{\substack{i=0 \\ i\leq\beta}}^{k} \theta(x - x_{\beta-i})\theta(x_{\beta-i+1} - x) \prod_{\substack{m=0 \\ m\neq i}}^{k} \frac{x - x_{\beta-i+m}}{x_\beta - x_{\beta-i+m}} \,. \tag{1.7}$$

Notice that the condition $i \leq \beta$ comes from the condition $\alpha \geq 0$. It is important to observe that the sum in Eq. (1.6) extends up to the $(N_x + k - 1)$-th node. Therefore, the original grid needs to be extended by $k - 1$ nodes. However, the range of validity of the interpolation remains that defined by the original grid, *i.e.* $x_0 \leq x \leq x_{N_x}$.

When implementing this interpolation procedure, it is important to realise that, typically, only a small number of terms in the sum in Eq. (1.6) is different from zero. For any given value of x, it is possible to determine the values of the index β for which the interpolating functions $w_\beta^{(k)}$ are different from zero, reducing (often dramatically) the amount of sums required to carry out an interpolation. The range of β is easily determined by observing that in Eq. (1.4) the summation extends on the nodes between x_α and $x_{\alpha+k}$. But since β is defined as $\alpha + i$, this exactly defines the range in β:

$$\alpha(x) \leq \beta \leq \alpha(x) + k \,, \tag{1.8}$$

where the function $\alpha(x)$ is implicitly defined through Eq. (1.1). Therefore, Eq. (1.6) becomes:

$$g(x) = \sum_{\beta=\alpha(x)}^{\alpha(x)+k} w_\beta^{(k)}(x) g(x_\beta) \,, \tag{1.9}$$

Since the interpolation functions $w_\beta^{(k)}(x)$ often appear inside integrals, it is very useful to use the fact that they are different from zero only over a limited interval, specifically:

$$w_\beta^{(k)}(x) \neq 0 \quad \Leftrightarrow \quad x_{\beta-k} < x < x_{\beta+1} \,. \tag{1.10}$$

This allows one to optimise the integration restricting the integration region only to where the interpolating functions are different from zero.

1.1 Generalised interpolation

In most of the applications within APFEL++ the assumption in Eq. (1.1) is used. However, sometimes it may be useful to release this assumption. A situation in which this is advantageous is in the presence of non-smooth or discontinuos functions (such as PDFs and FFs as functions of the factorisation scale μ in correspondence of the heavy-quark thresholds). When interpolating these functions one should not interpolate over the discontinuities. To do so and yet retain a given interpolation degree, one can release the assumption in Eq. (1.1). Specifically, we generalise it to:

$$x_{\alpha+t} < x \leq x_{\alpha+t+1} \quad \text{with} \quad t = 0, \ldots, k-1 \,, \tag{1.11}$$

such that the interpolation formula becomes:

$$g(x) = \sum_{\alpha=-t}^{N_x-t-1} \theta(x - x_{\alpha+t})\theta(x_{\alpha+t+1} - x) \sum_{i=0}^{k} g(x_{\alpha+i}) \prod_{\substack{m=0\\m\neq i}}^{k} \frac{x - x_{\alpha+m}}{x_{\alpha+i} - x_{\alpha+m}}\,, \tag{1.12}$$

that can be rearranged as:

$$g(x) = \sum_{\beta=-t}^{N_x+k-t-1} w_{\beta,t}^{(k)}(x) g(x_\beta)\,, \tag{1.13}$$

with:

$$w_{\beta,t}^{(k)}(x) = \sum_{i=0,i\leq\beta}^{k} \theta(x - x_{\beta-i+t})\theta(x_{\beta-i+t+1} - x) \prod_{m=0,m\neq i}^{k} \frac{x - x_{\beta-i+m}}{x_\beta - x_{\beta-i+m}}\,, \tag{1.14}$$

being the "generalised" interpolation functions. We observe that the support region of $w_{\beta,t}^{(k)}$ is:

$$w_{\beta,t}^{(k)}(x) \neq 0 \quad \Leftrightarrow \quad x_{\beta+t-k} < x < x_{\beta+t+1}\,, \tag{1.15}$$

that generalises Eq. (1.10). The generalised interpolation functions can be used to avoid interpolating over some particular grid nodes. In order to avoid interpolation over a specific node of the grid, one can choose t dynamically in such a way that $\beta + t$ in Eq. (1.14) never corresponds to that particular node. This mechanism is implemented in APFEL++ as follows. The interpolation grid is chosen to have two nodes in correspondence of the *threshold* x_T, but slightly displaced up and down by an "infinitesimal" amount ϵ to keep them separate, that is:

$$\{x_0, \ldots, x_{\alpha_t-1}, x_{\alpha_t}, \ldots, x_{N_x}\} \quad \text{with} \quad x_{\alpha_t-1} = x_T - \epsilon \quad \text{and} \quad x_{\alpha_t} = x_T + \epsilon\,. \tag{1.16}$$

The aim is then to avoid interpolating over the nodes x_{α_t-1} and x_{α_t}. By default we assume $t = 0$ in Eq. (1.14) so that we automatically reduce to Eq. (1.7). In this situation, we are assuming Eq. (1.1) where effectively the index α is determined dynamically according to the values of x. Therefore, we can effectively write:

$$x_{\alpha(x)} < x \leq x_{\alpha(x)+1}\,, \tag{1.17}$$

which implicitly defines the function $\alpha(x)$. Eq. (1.4) then requires summing over the $k + 1$ nodes of the grid $\{x_{\alpha(x)}, \ldots, x_{\alpha(x)+k}\}$. However, when the point x approaches x_T from below, the range $\{x_{\alpha(x)}, \ldots, x_{\alpha(x)+k}\}$ may end up enclosing both nodes x_{α_t-1} and x_{α_t}. To avoid this, we promote the index t in Eq. (1.11) to a function of x defined through the inequalities:

$$\begin{cases} x < x_T\,, \\ x_{\alpha_t-2} < x_{\alpha(x)-t(x)+k} \leq x_{\alpha_t-1}\,, \end{cases} \tag{1.18}$$

that translate into:

$$\begin{cases} \alpha(x) \leq \alpha_t - 2\,, \\ \alpha_t - 2 < \alpha(x) - t(x) + k \leq \alpha_t - 1\,. \end{cases} \tag{1.19}$$

Imposing the unnecessary but convenient constraint $t(x) \geq 0$, finally gives:

$$t(x) = \max\left[\min\left[\alpha(x), \alpha_t - 2\right] - \alpha_t + k + 1, 0\right]\,, \tag{1.20}$$

that also obeys:

$$0 \leq t(x) \leq k - 1\,, \tag{1.21}$$

as required. In addition, as in Eq. (1.9), the summation over β in Eq. (1.13) can be restricted to a range of $k + 1$ nodes as:

$$g(x) = \sum_{\beta=\alpha(x)-t(x)}^{\alpha(x)-t(x)+k} w_{\beta,t}^{(k)}(x) g(x_\beta)\,. \tag{1.22}$$

1.2 Bi-dimensional interpolation

As discussed in the section devoted to the computation of the convolution integrals, the interpolation functions can be used to compute integrals of the following kind:

$$I_1 = \int_{x_0}^{x_{N_x}} dx\, g(x) f(x)\,, \tag{1.23}$$

where f is a smooth function. Using Eqs. (1.6) and (1.10) we have that:

$$I_1 = \sum_{\beta=0}^{N_x+k-1} W_\beta g(x_\beta)\,, \tag{1.24}$$

with:

$$W_\beta = \int_{x_{\max(0,\beta-k)}}^{x_{\min(N_x,\beta+1)}} dx\, w_\beta^{(k)}(x) f(x)\,. \tag{1.25}$$

The equation above can be easily generalised to a bidimensional integral as:

$$I_2 = \int_{x_0}^{x_{N_x}} dx \int_{y_0}^{y_{N_y}} dy\, g(x,y) f(x,y) = \sum_{\alpha=0}^{N_x+k-1} \sum_{\beta=0}^{N_y+l-1} W_{\alpha\beta} g(x_\alpha, y_\beta)\,, \tag{1.26}$$

with:

$$W_{\alpha\beta} = \int_{x_{\max(0,\alpha-k)}}^{x_{\min(N_x,\alpha+1)}} dx \int_{y_{\max(0,\beta-k)}}^{y_{\min(N_y,\beta+1)}} dy\, w_\alpha^{(k)}(x)\, w_\beta^{(l)}(y)\, f(x,y)\,. \tag{1.27}$$

As discussed below in much detail, we mention that the functions w are piecewise. In particular, while they are continuous in correspondence of the nodes of the grid, their first derivative is not. As a consequence, the result of the numerical integrals in Eqs. (1.25) and (1.27) may be inaccurate. To overcome this problem, it is sufficient to split the integrals into sub-integrals over the intervals delimited by two consecutive nodes. Using Eq. (1.10), it is easy to see that, for an interpolation of degree k, one needs to compute $k+1$ integrals over the intervals included between the $(\beta-k)$-th and the $(\beta+1)$-th node.

1.3 Derivative and integral of an interpolated function

The simple form of the Lagrange polynomials allows, in some cases, for the analytic handling of operations such has derivation and integration of the interpolated functions. In this section we discuss how derivation and integration can be carried out analytically exploiting some specific properties of the Lagrange polynomials.

Referring Eq. (1.2), the main observation is that the interpolating functions $\ell_i^{(k)}$ are solutions of the following differential-equation system:

$$\begin{cases} \dfrac{d\ell_i^{(k)}}{dx} = \left(\displaystyle\sum_{\substack{n=0\\ n\neq i}}^{k} \frac{1}{x - x_{\alpha+n}} \right) \ell_i^{(k)}(x) \\ \\ \ell_i^{(k)}(x_{\alpha+i}) = 1 \end{cases}, \tag{1.28}$$

whose solution is Eq. (1.3). This allows us to compute the derivative of the test function g in Eq. (1.9) by only knowing its values on the grid points as:

$$\frac{dg}{dx} = \sum_{\beta=\alpha(x)}^{\alpha(x)+k} \mathcal{D}_\beta^{(k)}(x) g(x_\beta)\,, \tag{1.29}$$

where:

$$\mathcal{D}_\beta^{(k)}(x) = \sum_{i=0}^{\min(k,\beta)} \theta(x - x_{\beta-i})\theta(x_{\beta-i+1} - x) \left(\sum_{\substack{n=0\\ n\neq i}}^{k} \frac{1}{x_\beta - x_{\beta-i+n}} \prod_{\substack{m=0\\ m\neq i,n}}^{k} \frac{x - x_{\beta-i+m}}{x_\beta - x_{\beta-i+m}} \right)\,. \tag{1.30}$$

It should be pointed out that, due to the discontinuity of the derivative of the interpolation functions on the grid nodes, numerical derivation through Eq. (1.29) does not work when x coincides with a grid node. This problem can be overcome by simply slightly displacing the value of x in case it falls on a grid node.

Eq. (1.28) can also be used to compute integrals of the function g. In particular, suppose we want to compute the integral:

$$I(a,b) = \int_a^b dy\, g(y)\,. \tag{1.31}$$

Using Eq. (1.9), we find that:

$$I(a,b) = \int_a^b dy \sum_{\beta=\alpha(y)}^{\alpha(y)+k} w_\beta^{(k)}(y) g(x_\beta) = \int_a^b dy \left[w_{\alpha(y)}^{(k)}(y) g(x_{\alpha(y)}) + \cdots + w_{\alpha(y)+k}^{(k)}(y) g(x_{\alpha(y)+k}) \right]\,. \tag{1.32}$$

The dependence on the integration variable y of the index α complicates the solution of this integral. However, it is easy to derive a close form for the function $\alpha(y)$:

$$\alpha(y) = -1 + \sum_{\beta=0}^{N_x} \theta(y - x_\beta)\,, \tag{1.33}$$

that (obviously) means that $\alpha(y)$ is constant on the separate intervals $[x_0 : x_1]$, $[x_1 : x_2]$, and so on. This allows us to break the integral above as follows:

$$\int_a^b dy \sum_{\beta=\alpha(y)}^{\alpha(y)+k} w_\beta^{(k)}(y) g(x_\beta) = \left[\int_a^{x_{\alpha(a)+1}} + \sum_{\gamma=\alpha(a)+1}^{\alpha(b)-1} \int_{x_\gamma}^{x_{\gamma+1}} + \int_{x_{\alpha(b)}}^b \right] dy \sum_{\beta=\alpha(y)}^{\alpha(y)+k} w_\beta^{(k)}(y) g(x_\beta)\,, \tag{1.34}$$

such that in each single integral the integrand has a constant value of α. The three terms above can be separately rearranged as:

$$\sum_{\beta=\alpha(a)}^{\alpha(a)+k} g(x_\beta) \int_a^{x_{\alpha(a)+1}} dy\, w_\beta^{(k)}(y)\,, \tag{1.35}$$

$$\sum_{\gamma=\alpha(a)+1}^{\alpha(b)-1} \sum_{\beta=\gamma}^{\gamma+k} g(x_\beta) \int_{x_\gamma}^{x_{\gamma+1}} dy\, w_\beta^{(k)}(y)\,, \tag{1.36}$$

$$\sum_{\beta=\alpha(b)}^{\alpha(b)+k} g(x_\beta) \int_{x_{\alpha(b)}}^{b} dy\, w_\beta^{(k)}(y)\,, \tag{1.37}$$

where we have used the fact that $\alpha(x_\gamma) = \gamma$. Notice that with this we managed to obtain direct integrations of the interpolating functions $w_\beta^{(k)}$. Now we note that the three terms above span the range of nodes $\beta \in [\alpha(a) : \alpha(b) + k]$. Therefore, the integral in Eq. (1.31) can be written as:

$$I(a,b) = \sum_{\beta=\alpha(a)}^{\alpha(b)+k} \mathcal{G}_\beta^{(k)}(a,b) g(x_\beta)\,, \tag{1.38}$$

with:

$$\begin{aligned} \mathcal{G}_\beta^{(k)}(a,b) &= \int_{\max(a,x_{\beta-k})}^{\min(b,x_{\beta+1})} dy\, w_\beta^{(k)}(y) \\ &= \sum_{i=0}^{\min(k,\beta)} \int_{\max(a,x_{\beta-k})}^{\min(b,x_{\beta+1})} dy\, \theta(y - x_{\beta-i}) \theta(x_{\beta-i+1} - y) \prod_{\substack{m=0 \\ m\neq i}}^{k} \frac{y - x_{\beta-i+m}}{x_\beta - x_{\beta-i+m}}\,. \end{aligned} \tag{1.39}$$

Now, assuming that $a \leq b$, we can write the integral above as:

$$\int_{\max(a,x_{\beta-k})}^{\min(b,x_{\beta+1})} dy\, \theta(y - x_{\beta-i}) \theta(x_{\beta-i+1} - y) \cdots = \theta(b - x_{\beta-i}) \theta(x_{\beta-i+1} - a) \int_{\max(a,x_{\beta-i})}^{\min(b,x_{\beta-i+1})} dy \ldots\,, \tag{1.40}$$

so that:

$$\mathcal{G}_\beta^{(k)}(a,b) = \sum_{i=0}^{\min(k,\beta)} \theta(b - x_{\beta-i})\theta(x_{\beta-i+1} - a) \prod_{\substack{m=0\\m\neq i}}^{k} \frac{1}{x_\beta - x_{\beta-i+m}} \int_{\max(a,x_{\beta-i})}^{\min(b,x_{\beta-i+1})} dy \prod_{\substack{m=0\\m\neq i}}^{k} (y - x_{\beta-i+m}) \,. \tag{1.41}$$

It is easy to see that:

$$\prod_{\substack{m=0\\m\neq i}}^{k} (y - x_{\beta-i+m}) = \sum_{n=0}^{k} (-1)^n p_\beta^{(k)}(n) y^{k-n} \,, \tag{1.42}$$

such that:

$$\begin{aligned} \mathcal{G}_\beta^{(k)}(a,b) &= \sum_{i=0}^{\min(k,\beta)} \theta(b - x_{\beta-i})\theta(x_{\beta-i+1} - a) \\ &\times \left[\prod_{\substack{m=0\\m\neq i}}^{k} \frac{1}{x_\beta - x_{\beta-i+m}} \right] \sum_{n=0}^{k} \frac{(-1)^n p_\beta^{(k)}(n)}{k-n+1} \left(\overline{b}^{k-n+1} - \overline{a}^{k-n+1} \right) \,. \end{aligned} \tag{1.43}$$

with $\overline{a} = \max(a, x_{\beta-i})$ and $\overline{b} = \min(b, x_{\beta-i+1})$. What is left to do is to determine the coefficients $p_\beta^{(k)}$. For convenience, let us define the set $\{r_1, \ldots, r_k\} = \{x_{\beta-i}, \ldots, x_{\beta-1}, x_{\beta+1}, \ldots, x_{\beta-i+k}\}$.[2] The coefficients $p_\beta^{(k)}$ defined in Eq. (1.42) can be expressed as:

$$p_\beta^{(k)}(n) = \sum_{i_1=1}^{k} r_{i_1} \sum_{i_2=i_1+1}^{k} r_{i_2} \cdots \sum_{i_n=i_{n-1}+1}^{k} r_{i_n} \,. \tag{1.44}$$

In order to obtain a convenient algorithm to compute the coefficients $p_\beta^{(k)}$, we define the vectorial function:

$$\mathbf{f}^{(k)}(\mathbf{r}, \mathbf{a}) \quad \text{with components} \quad f_j^{(k)}(\mathbf{r}, \mathbf{a}) = \sum_{i=j+1}^{k} r_i a_i \,. \tag{1.45}$$

It is important to notice that while $\mathbf{r}$ is a k-dimensional vector with index running between 1 and k, $\mathbf{a}$ and $\mathbf{f}^{(k)}$ are in principle infinite-dimensional vectors with index running between $-\infty$ and $+\infty$. However, given the definition of its components in Eq. (1.44), it turns out that $f_j^{(k)} = 0$ for $j \geq k$. The same applies to $\mathbf{a}$ because, as we will see below, it has to be identified with some $\mathbf{f}^{(k)}$. The function $\mathbf{f}^{(k)}$ can now be used to define the vector $\mathbf{P}^{(k)}(n)$ recursively. The relevant recursive relation is:

$$\mathbf{P}^{(k)}(n+1) = \mathbf{f}^{(k)}(\mathbf{r}, \mathbf{P}^{(k)}(n)) \quad \text{with} \quad \mathbf{P}^{(k)}(0) = \mathbf{1} \,. \tag{1.46}$$

Finally, the coefficient $p_\beta^{(k)}(n)$ is the zero-th component of the vector $\mathbf{P}^{(k)}(n)$, *i.e.* $p_\beta^{(k)}(n) \equiv P_0^{(k)}(n)$.

Now, we turn to derive derivation and integration formulas for the generalised interpolation formula in Eq. (1.22). The derivative is as simple as in the $t = 0$ case, that is:

$$\frac{dg}{dx} = \sum_{\beta=\alpha(x)-t(x)}^{\alpha(x)-t(x)+k} \mathcal{D}_{\beta,t}^{(k)}(x) g(x_\beta) \,, \tag{1.47}$$

with:

$$\mathcal{D}_{\beta,t}^{(k)}(x) = \sum_{i=0}^{\min(k,\beta)} \theta(x - x_{\beta-i+t(x)})\theta(x_{\beta-i+t(x)+1} - x) \left(\sum_{\substack{n=0\\n\neq i}}^{k} \frac{1}{x_\beta - x_{\beta-i+n}} \prod_{\substack{m=0\\m\neq i,n}}^{k} \frac{x - x_{\beta-i+m}}{x_\beta - x_{\beta-i+m}} \right) \,. \tag{1.48}$$

[2] From the definition of the set $\{r_1, \ldots, r_k\}$, it becomes clear that the subscript β of the coefficients $p_\beta^{(k)}$ refers to the fact that this coefficients are computed on a vector of k nodes where the node x_β has been removed.

Due to the presence of the function $t(x)$, defined in Eq. (1.20), the integration procedure is more involved. We want to compute:

$$\begin{aligned} I(a,b) &= \int_a^b dy \sum_{\beta=\alpha(y)-t(y)}^{\alpha(y)-t(y)+k} w_{\beta,t}^{(k)}(y) g(x_\beta) \\ &= \left[\int_a^{x_{\alpha(a)+1}} + \sum_{\gamma=\alpha(a)+1}^{\alpha(b)} \int_{x_\gamma}^{x_{\gamma+1}} - \int_b^{x_{\alpha(b)+1}} \right] dy \sum_{\beta=\alpha(y)-t(y)}^{\alpha(y)-t(y)+k} w_{\beta,t}^{(k)}(y) g(x_\beta) \,. \end{aligned} \tag{1.49}$$

As in the case of $t = 0$, each of the integrals above has a constant value of $\alpha(y)$ and $t(y)$ so that sum over β and integral signs can be exchanged:

$$\begin{aligned} I(a,b) &= \sum_{\beta=\alpha(a)-t(a)}^{\alpha(a)-t(a)+k} g(x_\beta) \int_a^{x_{\alpha(a)+1}} dy\, w_{\beta,t}^{(k)}(y) \\ &+ \sum_{\gamma=\alpha(a)+1}^{\alpha(b)} \sum_{\beta=\gamma-t_\gamma}^{\gamma-t_\gamma+k} g(x_\beta) \int_{x_\gamma}^{x_{\gamma+1}} dy\, w_{\beta,t}^{(k)}(y) \\ &- \sum_{\beta=\alpha(b)-t(b)}^{\alpha(b)-t(b)+k} g(x_\beta) \int_b^{x_{\alpha(b)+1}} dy\, w_{\beta,t}^{(k)}(y) \,, \end{aligned} \tag{1.50}$$

with:

$$t_\gamma = t(x_\gamma) = \max\left[\min\left[\gamma, \alpha_t - 2\right] - \alpha_t + k + 1, 0\right] \,. \tag{1.51}$$

It turns out to be very complicated to write the expression in Eq. (1.50) in a more compact form, such as that in Eq. (1.38). Therefore, we implement Eq. (1.50) as it is. The one thing left to do is to solve the integrals. This is done by using the general formula:

$$\begin{aligned} \int_{\ell_1}^{\ell_2} dy\, w_{\beta,t}^{(k)}(y) &= \sum_{i=0}^{\min(k,\beta)} \theta(\ell_2 - x_{\beta-i+t_\ell}) \theta(x_{\beta-i+t_\ell+1} - \ell_1) \\ &\times \left[\prod_{\substack{m=0 \\ m\neq i}}^{k} \frac{1}{x_\beta - x_{\beta-i+m}} \right] \sum_{n=0}^{k} \frac{(-1)^n p_\beta^{(k)}(n)}{k-n+1} \left(\bar{\ell}_2^{k-n+1} - \bar{\ell}_1^{k-n+1} \right) \,, \end{aligned} \tag{1.52}$$

with $t(\ell_1) = t(\ell_2) = t_\ell$, and $\bar{\ell}_1 = \max(\ell_1, x_{\beta-i+t_\ell})$ and $\bar{\ell}_2 = \min(\ell_2, x_{\beta-i+t_\ell+1})$.