Neutron star cooling by axion emission

Ray Hagimoto

August 8, 2023

Contents

1 Introduction

The goal of this project is to evaluate the integral

$$\mathrm{d}\varepsilon_{3'} = f_1 f_2 (1 - f_{1'})(1 - f_{2'})(1 + f_{3'})\, E_{3'} \sum_{\sigma,\sigma'} |\mathcal{M}|^2 \, \mathrm{d}\Phi_n(1, 2; 1', 2', 3') \tag{1.1}$$

where,

$$\mathrm{d}\Phi_n\big((p_a + p_b)^2; 1, \ldots, n\big) \equiv \prod_{i=1}^{n} \left[\delta(p_i^2 - m_i^2)\,\Theta(p_i^0)\,\frac{\mathrm{d}^4 p_i}{(2\pi)^3}\right] (2\pi)^4 \delta^4(p_a + p_b - p_1 - \cdots - p_n)$$

is the Lorentz-invariant phase space measure, and

$$f_i \equiv f_{FD}(E_i) \equiv \frac{1}{e^{(E_i - \mu_i)/T} + 1}$$

is the Fermi-Dirac distribution. The spin-summed matrix element squared is given by

$$\sum_{\sigma,\sigma'} |\mathcal{M}|^2 = \frac{128\, g_{ae\mu}^2 e^4}{(m_1^2 - m_{1'}^2)^2} \, \frac{(p_1 \cdot p_{1'} - m_1 m_{1'})(p_2 \cdot p_{3'})(p_{2'} \cdot p_{3'})}{(p_2 - p_{2'})^4} \ . \tag{1.2}$$

We can use the momentum conserving Dirac delta to evaluate the $\boldsymbol{p}_{2'}$ integral, setting

$$\boldsymbol{p}_{2'} = \boldsymbol{p}_1 + \boldsymbol{p}_2 - \boldsymbol{p}_{1'} - \boldsymbol{p}_{3'}. \tag{1.3}$$

We now choose to align the z-axis with $\boldsymbol{p}_1$ and measure angles with respect to this axis. Converting to spherical polar coordinates gives, for example, $\mathrm{d}^3 p_{3'} = p_{3'}^2 \, \mathrm{d}p_{3'} \, \mathrm{d}\cos\theta_{13'} \, \mathrm{d}\phi_{13'}$ * . Then the energy conserving Dirac delta can be used to evaluate the $\mathrm{d}p_{3'}$ integral in the following way. First, we use $E = \sqrt{p^2 + m^2}$ to rewrite the masses. Then we use momentum conservation to make the replacement $\boldsymbol{p}_{2'} \to \boldsymbol{p}_1 + \boldsymbol{p}_2 - \boldsymbol{p}_{1'} - \boldsymbol{p}_{3'}$:

$$\begin{aligned}
&E_1 + E_2 = E_{1'} + E_{2'} + E_{3'} \\
&\Rightarrow \sqrt{p_1^2 + m_1^2} + \sqrt{p_2^2 + m_2^2} = \sqrt{p_{1'}^2 + m_{1'}^2} + \sqrt{p_{2'}^2 + m_{2'}^2} + \sqrt{p_{3'}^2 + m_{3'}^2} \\
&\Rightarrow \sqrt{p_1^2 + m_1^2} + \sqrt{p_2^2 + m_2^2} = \sqrt{p_{1'}^2 + m_{1'}^2} + \sqrt{(\boldsymbol{p}_1 + \boldsymbol{p}_2 - \boldsymbol{p}_{1'} - \boldsymbol{p}_{3'})^2 + m_{2'}^2} + \sqrt{p_{3'}^2 + m_{3'}^2}.
\end{aligned} \tag{1.4}$$

We would like to solve eqn. (1.4) for $p_{3'}$, but there is a problem coming from the fact that $p_{3'}$ appears under two square root symbols which makes it impossible to get an expression of the form $p_{3'} = \cdots$. Instead, we will make the approximation that because $p_{3'} \ll p_i$ for $i \in \{1, 2, 1', 2'\}$ that we can ignore the $p_{3'}$ that shows up in $\sqrt{(\boldsymbol{p}_1 + \boldsymbol{p}_2 - \boldsymbol{p}_{1'} - \boldsymbol{p}_{3'})^2 + m_{2'}^2}$.

*The angle $\phi_{13'}$ is not measured with respect to $\boldsymbol{p}_1$, it is measured to some axis orthogonal to $\boldsymbol{p}_1$. We don't need to define that axis explicitly as long as the other angles ϕ_{1i} are measured with respect to the same axis.

The result we obtain when solving for $p_{3'}$ is then

$$\begin{aligned} p_{3'} = \Big[& m_1^2 + m_{1'}^2 + m_2^2 + m_{2'}^2 - m_{3'}^2 + 2p_1^2 - 2p_1 p_{1'} c_{11'} + 2p_{1'}^2 - 2E_1 E_{1'} \\ & + 2p_1 p_2 c_{12} - 2p_2 p_{1'} c_{21'} + 2p_2^2 + 2E_1 - 2E_{1'} E_2 \\ & - 2E_1 E_{2'}(p_1, p_2, p_{1'}, c_{11'}, c_{12}, c_{21'}) + 2E_{1'} E_{2'}(p_1, p_2, p_{1'}, c_{11'}, c_{12}, c_{21'}) \\ & - 2E_2 E_{2'}(p_1, p_2, p_{1'}, c_{11'}, c_{12}, c_{21'}) \Big]^{1/2} \end{aligned} \tag{1.5}$$

where $E_i \equiv \sqrt{p_i^2 + m_i^2}$, $c_{ij} \equiv \cos\theta_{ij} \equiv \boldsymbol{p}_i \cdot \boldsymbol{p}_j / (p_i p_j)$ and

$$E_{2'}(p_1, p_2, p_{1'}, c_{11'}, c_{12}, c_{21'}) = \sqrt{p_1^2 - 2p_1 p_{1'} c_{11'} + p_{1'}^2 + 2p_1 p_2 c_{12} - 2p_2 p_{1'} c_{21'} + p_2^2 + m_{2'}^2}. \tag{1.6}$$

Now that we have an expression for $p_{3'}$ the requirement that $p_{3'}$ must be real and non-negative restricts the domain of integration of the other variables. I am not sure how to derive the new limits so I got stuck here.

1.1 A recursive expression for phase space integrals

The following regurgitates parts of [1, 2]. The problem I ran into at the end of the last section was how to use the energy-momentum delta functions to restrict the bounds of integration. Ref. [1] provides a means to circumvent this problem by performing a change of variables. Consider the phase space integral $R_n(p_a + p_b)$ defined by (where p_i now denotes a four-momentum vector in constrast to the previous section where it represented a three-momentum magnitude.)

$$R_n(p_a + p_b) = \int \cdots \int \prod_{i=1}^{n} \delta(p_i^2 - m_i^2)\, \Theta(p_i^0)\, \mathrm{d}^4 p_i\, \delta^4(p_a + p_b - p_1 - \cdots - p_n). \tag{1.7}$$

This is manifestly Lorentz invariant and therefore R_n may only be a function of $s \equiv (p_a + p_b)^2$. Define the new integration variable $M_{n-1}^2 \equiv (p_a + p_b - p_n)^2$. The physical significance of M_{n-1} is that it is the invariant mass of the first $n-1$ particles, which can be seen using four-momentum conservation: $p_a + p_b - p_n = p_1 + p_2 + \cdots + p_{n-1}$. It is possible to show that the following recursive relation holds (eqn. (3) of [1])

$$R_n(s) = \int_{(m_1 + m_2 + \cdots + m_{n-1})^2}^{(\sqrt{s} - m_n)^2} \mathrm{d}M_{n-1}^2 \int \mathrm{d}\Omega_n \frac{\sqrt{\lambda(s, M_{n-1}^2, m_n^2)}}{8s} R_{n-1}(M_{n-1}^2) \tag{1.8}$$

where $\lambda(x, y, z) = x^2 + y^2 + z^2 - 2xy - 2xz - 2yz$ and the solid angle $\mathrm{d}\Omega_n \equiv \mathrm{d}\cos\theta_n\, \mathrm{d}\varphi_n$ defines the direction of $\boldsymbol{p}_n$ in the frame where $p_a + p_b = (\sqrt{s}, \mathbf{0})$. The upper limit is obtained by requiring that

$$|\boldsymbol{p}_n|^2 = \frac{\lambda(s, M_{n-1}^2, m_n^2)}{4s} \tag{1.9}$$

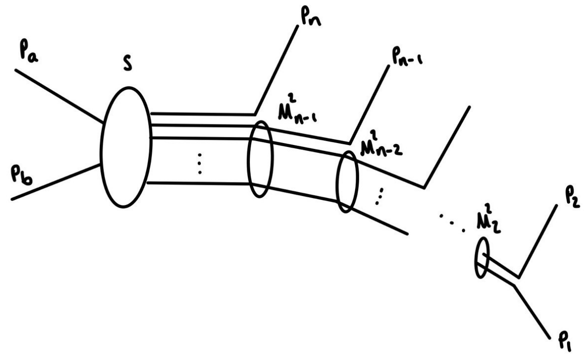

Figure 1: Illustration of the recursion relation as a sequence of effective $2 \to 2$ scattering events.

be positive whereas the lower limit is threshold below which $R_{n-1}(M^2_{n-1}) = 0^\dagger$. Repeated application of eq. (1.8) yields

$$\begin{aligned} R_n(s) = &\int_{(m_1+m_2+\cdots+m_{n-1})^2}^{(\sqrt{s}-m_n)^2} \mathrm{d}M^2_{n-1} \int \mathrm{d}\Omega_n \frac{\sqrt{\lambda(s, M^2_{n-1}, m^2_n)}}{8s} \\ &\times \int_{(m_1+m_2+\cdots+m_{n-2})^2}^{(M_{n-1}-m_{n-1})^2} \mathrm{d}M^2_{n-2} \int \mathrm{d}\Omega_{n-1} \frac{\sqrt{\lambda(M^2_{n-1}, M^2_{n-2}, m^2_{n-1})}}{8M^2_{n-1}} \\ &\times \cdots \times \int_{(m_1+m_2)^2}^{(M_3-m_3)^2} \mathrm{d}M^2_2 \int \mathrm{d}\Omega_3 \frac{\sqrt{\lambda(M^2_3, M^2_2, m^2_3)}}{8M^2_3} \times \int \mathrm{d}\Omega_2 \frac{\sqrt{\lambda(M^2_2, m^2_1, m^2_2)}}{8M^2_2}. \end{aligned} \tag{1.10}$$

The important case for me is $n = 3$:

$$R_3(s) = \int_{(m_1+m_2)^2}^{(\sqrt{s}-m_3)^2} \mathrm{d}M^2_2 \int \mathrm{d}\Omega_3 \, \frac{\lambda(s, M^2_2, m^2_3)}{8s} \int \mathrm{d}\Omega_2 \, \frac{\lambda(M^2_2, m^2_1, m^2_2)}{8M^2_2} \, . \tag{1.11}$$

It's easy to verify that the number of integration variables matches our expectation of $3(3) - 4 = 5$ since we are integrating over one invariant mass, and two pairs of two angles.

In the above discussion we have considered p_a and p_b to be fixed. However, for my purposes I will also be integrating over p_a and p_b. This case seems to be similar to the one considered in [2] (cf. the example given between their eqs. (12) and (13)).

1.2 Writing integral in terms of momentum transfers

In sec 1.1 we wrote the phase space integral in terms of invariant masses and angles of the three-momenta $\boldsymbol{p}_i$ defined in the center-of-mass frame where $\sum_{k=1}^{i} p_k = (M_i, \mathbf{0})$. The matrix

†In eq. (3) of [1] the authors allude that the angular $\mathrm{d}\Omega$ integrals can be performed. They do not perform the integrals because *"the variables defining a momentum vector should appear explicitly if a Monte Carlo method is to be applied for the generation of events."* In my case I think that we still can't do these integrals trivially because the matrix element depends on them, although I'm not 100% sure if this is correct.

element in eq. (1.2) is a function of the Lorentz invariants $(p_a \cdot p_1)$, $(p_b \cdot p_3)$, $(p_2 \cdot p_3)$, and $(p_b \cdot p_2) = (-(p_a - p_1 - p_2 - p_3) \cdot p_2)$. It is cumbersome to write this matrix element explicitly in terms of the angles $\{(\theta_i, \varphi_i) : i = 1, 2, \ldots, n\}$ so we would rather use kinematic Lorentz invariants analogous to the Mandelstam variables for $2 \to 2$ scattering. To this end it is convenient to introduce the so-called 'momentum transfers' $t_i \equiv Q_i^2 \equiv (p_a - p_1 - \cdots - p_i)^2 = (p_n + p_{n-1} + \cdots + p_{i+1} - p_b)^2$.

Some of the dot products may be written exclusively in terms of the kinematic variables. For example,

$$
\begin{aligned}
t_1 &\equiv (p_a - p_1)^2 & &\to & 2p_a \cdot p_1 &= m_a^2 + m_1^2 - t_1 \\
t_2 &\equiv (p_a - (p_1 + p_2))^2 & &\to & 2p_a \cdot p_2 &= M_2^2 - m_1^2 + t_1 - t_2 \\
t_3 &\equiv (p_a - (p_1 + p_2 + p_3))^2 & &\to & 2p_a \cdot p_3 &= M_3^2 - M_2^2 + t_2 \\
M_2^2 &\equiv (p_1 + p_2)^2 & &\to & 2p_1 \cdot p_2 &= M_2^2 - m_1^2 - m_2^2 \\
M_3^2 &\equiv (p_1 + p_2 + p_3)^2 & &\to & 2(p_1 + p_2) \cdot p_3 &= M_3^2 - 2M_2^2 + m_1^2 + m_2^2 - m_3^2
\end{aligned}
$$

However I was unable to write $p_2 \cdot p_3$ in terms of the invariants alone. $p_b \cdot p_2$ is really just $p_2 \cdot p_3$ in disguise so it has the same problem. I believe this is because these variables are not enough to fully describe the system and some angles are required in addition. We need $3(3) - 4 = 5$ integration variables but only 3 of them are prescribed by the kinematic invariants M_2^2, t_1, and t_2[‡]. I haven't followed their derivation through in enough detail to understand precisely how they show up in the expression .

Understand how to write the matrix element explicitly in terms of the angles φ_1, φ_2 as well as the invariants M_2^2, t_1, and t_2.

The two additional degrees of freedom can be seen in eq. (14) of [1]. They are the azimuthal angles φ_1, φ_2 as seen in a frame where $p_a = (m_a, \mathbf{0})$.

2 Monte Carlo integration

2.1 The basics

Suppose you are interested in evaluating the integral

$$I = \int_a^b f(x)\,\mathrm{d}x. \tag{2.1}$$

The most obvious method of approximating this is with a Riemann sum or trapezoidal integration where you divide the domain into bins of width Δx and calculate the area of a rectangle or trapezoid using the values of f at the bin edges.

Monte Carlo integration, on the other hand, takes a a different approach altogether whereby points in the domain are sampled at random and the integral is approximated as an average. To see this, note that the average value of f on the domain is

$$\langle f \rangle = \frac{1}{b-a} \int_a^b f(x)\,\mathrm{d}x = \frac{I}{b-a}. \tag{2.2}$$

[‡] $M_1^2 = m_1^2$, $M_3^2 = s$ and $t_3 = (-p_b)^2 = m_b^2$ are all constants, not integration variables.

The average $\langle f \rangle$ can be approximated by uniformly sampling N i.i.d random numbers x_i on the domain $x \in [a, b]$ where $i \in 1, 2, ..., N$ and calculating the mean of $f(x_i)$:

$$\langle f \rangle \approx \langle f \rangle_N \equiv \frac{1}{N} \sum_{i=1}^{N} f(x_i). \tag{2.3}$$

Therefore we can approximate the integral I as

$$I \approx (b - a)\langle f \rangle_N. \tag{2.4}$$

This argument can be extended to higher dimensions, i.e. in n dimensions we have

$$I = \int_V f(\boldsymbol{x}) \, \mathrm{d}^n x = V \langle f \rangle. \tag{2.5}$$

One important result is that this strategy always converges as $N^{-1/2}$ in all dimensions, whereas techniques like trapezoidal integration converge rapidly in one dimension (N^{-2}), but significantly slower for higher dimensions ($N^{-2/n}$). For example in 5 dimensions, trapezoidal integration converges like $N^{-2/5}$, which is slower than Monte Carlo integration.

2.2 Importance sampling

In sec. 2.1 we sampled x uniformly at random on the domain of integration. However, we can choose to sample x from any arbitrary distribution $p(x)$ and compute the same expectation value by weighting the sample $f(x)$ by $1/p(x)$ so that

$$I \approx \frac{V}{N} \sum_{i=1}^{N} \frac{f(x_i)}{p(x_i)}. \tag{2.6}$$

With an appropriate choice of p our Monte Carlo integration technique can be made to converge significantly faster than by sampling x uniformly at random. To see why this is, suppose $f(x) = e^{-x^2}$ and $[a, b] = [-1000, 1000]$. Sampling uniformly on $[-1000, 1000]$ means that the majority of points we pick will be in the tails of the Gaussian and will hardly affect the value of the integral. If instead we sampled from a standard normal distribution we would mostly choose x values near the peak at $x = 0$ and our result would converge much more rapidly. Clearly the choice of p is heavily dependent on the particular shape of the integrand f. The Metropolis-Hastings algorithm is one way to generate samples x from $p(x)$ and can be used for importance sampling.

<u>VEGAS algorithm:</u>

The discussion of the VEGAS algorithm at `https://www.ippp.dur.ac.uk/~krauss/Lectures/QuarksLeptons/Basics/PS_Vegas.html` is quite good. In case the link breaks in the future I reproduce some of the main points below.

> "The VEGAS algorithm starts by dividing the n-dimensional hypercube $[0, 1]^n$ into smaller ones – usually of identical size – and performs a sampling in each of them. The results are then used to refine the integration grid for the next iteration of the sampling procedure."

A Derivation of recursive relation

This appendix fills in the details for the derivation of eq. (1.8) which are left out in [1]. Our starting point is

$$
\begin{aligned}
R_n(s) &= \int \prod_{i=1}^{n} \delta(p_i^2 - m_i^2)\,\Theta(p_i^0)\,\mathrm{d}^4 p_i\,\delta^4(p_a + p_b - p_1 - \cdots - p_n) \\
&= \int \delta(p_n^2 - m_n^2)\,\Theta(p_n^0)\,\mathrm{d}^4 p_n \left[\prod_{i=1}^{n-1} \delta(p_i^2 - m_i^2)\,\Theta(p_i^0)\right] \delta^4(p_a + p_b - p_n - p_1 - \cdots - p_{n-1}) \\
&= \int \delta(p_n^2 - m_n^2)\,\Theta(p_n^0)\,\mathrm{d}^4 p_n\, R_{n-1}(p_a + p_b - p_n)
\end{aligned}
$$

Multiply the RHS by $\int \mathrm{d}M_{n-1}^2\,\delta(M_{n-1}^2 - (p_a + p_b - p_n)^2)$.

$$
\int \mathrm{d}M_{n-1}^2\,\delta\big(M_{n-1}^2 - (p_a + p_b - p_n)^2\big)\,\delta(p_n^2 - m_n^2)\,\Theta(p_n^0)\,\mathrm{d}^4 p_n\, R_{n-1}\big(p_a + p_b - p_n\big).
$$

Use the on-shell delta function to do the p_n^0 integral and write $\mathrm{d}^3 p_n$ in terms of spherical coordinates:

$$
\begin{aligned}
&\int \mathrm{d}M_{n-1}^2\,\delta\big(M_{n-1}^2 - (p_a + p_b - p_n)^2\big)\,\frac{\mathrm{d}^3 p_n}{2E_n}\,R_{n-1}\big(p_a + p_b - p_n\big) \\
&= \int \mathrm{d}M_{n-1}^2\,\delta\big(M_{n-1}^2 - (p_a + p_b - p_n)^2\big)\,\frac{|\boldsymbol{p}_n|^2 \mathrm{d}|\boldsymbol{p}_n| \mathrm{d}\Omega_n}{2E_n}\,R_{n-1}\big(p_a + p_b - p_n\big).
\end{aligned}
$$

Then use the M_{n-1}^2 Dirac delta to do the $\mathrm{d}|\boldsymbol{p}_n|$ integral. This requires us to (i) solve $M_{n-1}^2 - (p_a + p_b - p_n)^2 = 0$ for $|\boldsymbol{p}_n|$; (ii) evaluate $\mathrm{d}/\mathrm{d}|\boldsymbol{p}_n|\big(M_{n-1}^2 - (p_a + p_b - p_n)^2\big)$; and (iii) impose $|\boldsymbol{p}_n|^2 \geq 0$ to derive bounds of integration for $\int \mathrm{d}M_{n-1}^2$.

(i) We start by expanding the LHS

$$
\begin{aligned}
&M_{n-1}^2 - (p_a + p_b - p_n)^2 = M_{n-1}^2 - (p_a + p_b)^2 - p_n^2 + 2(p_a + p_b)\cdot p_n \\
&= M_{n-1}^2 - s - m_n^2 + 2(E_a + E_b)E_n + 2(\boldsymbol{p}_a + \boldsymbol{p}_b)\cdot \boldsymbol{p}_n.
\end{aligned}
$$

Now we choose the center of mass frame so that $\boldsymbol{p}_a + \boldsymbol{p}_b = 0$. This is valid because the integral is a Lorentz invariant, so the final result is independent of the reference frame.

$$
\begin{aligned}
&M_{n-1}^2 - s - m_n^2 + 2(E_a + E_b)E_n + 2(\boldsymbol{p}_a + \boldsymbol{p}_b)\cdot \boldsymbol{p}_n \\
&= M_{n-1}^2 - s - m_n^2 + 2\sqrt{s}\,\sqrt{|\boldsymbol{p}_n|^2 + m_n^2} = 0.
\end{aligned}
$$

Now we can rearrange and solve for $|\boldsymbol{p}_n|$

$$
\begin{aligned}
|\boldsymbol{p}_n|^2 &= \frac{(s + m_n^2 - M_{n-1}^2)^2}{4s} - m_n^2 = \frac{s^2 + m_n^4 + M_{n-1}^4 - 2sm_n^2 - 2sM_{n-1}^2 - 2m_n^2 M_{n-1}^2}{4s} \\
&\equiv \boxed{\frac{\lambda(s, M_{n-1}^2, m_n^2)}{4s}}\,.
\end{aligned}
$$

$\lambda(x, y, z) = x^2 + y^2 + z^2 - 2xy - 2xz - 2yz$ is called the Källen function.

(ii)

$$
\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}|\boldsymbol{p}_n|}\left(M_{n-1}^2-(p_a+p_b-p_n)^2\right)\\
&=\frac{\mathrm{d}}{\mathrm{d}|\boldsymbol{p}_n|}\left(M_{n-1}^2-s-m_n^2+2\sqrt{s}\sqrt{|\boldsymbol{p}_n|^2+m_n^2}\right)\\
&=\boxed{\frac{2\sqrt{s}\,|\boldsymbol{p}_n|}{E_n}}
\end{aligned}
$$

(iii)

$$
\begin{aligned}
&|\boldsymbol{p}_n|^2\geq 0\\
&\Rightarrow \lambda(s,M_{n-1}^2,m_n^2)\geq 0\\
&\Rightarrow \left[M_{n-1}^2-(s+m_n^2)\right]^2-4\,sm_n^2\geq 0\\
&\Rightarrow \left[M_{n-1}^2-(s+m_n^2)\right]^2\geq 4\,sm_n^2\\
&\Rightarrow M_{n-1}^2-(s+m_n^2)\geq 2\sqrt{s\,m_n^2}\;\big|\big|\;M_{n-1}^2-(s+m_n^2)\leq -2\sqrt{s\,m_n^2}\\
&\Rightarrow M_{n-1}^2\geq (s+m_n^2)+2\sqrt{s\,m_n^2}\;\big|\big|\;M_{n-1}^2\leq (s+m_n^2)-2\sqrt{s\,m_n^2}\\
&\Rightarrow \boxed{M_{n-1}^2\geq(\sqrt{s}+m_n)^2\;\big|\big|\;M_{n-1}^2\leq(\sqrt{s}-m_n)^2.}
\end{aligned}
$$

With these results established we find that

$$
\delta\big(M_{n-1}^2-(p_a+p_b-p_n)^2\big)=\frac{\delta\Big(|\boldsymbol{p}_n|-\sqrt{\lambda(s,M_{n-1}^2,m_n^2)}\Big/(2\sqrt{s})\Big)}{2\sqrt{s}\,|\boldsymbol{p}_n|/E_n}\quad. \tag{1.1}
$$

Hence,

$$
\begin{aligned}
&\int \mathrm{d}M_{n-1}^2\,\delta\big(M_{n-1}^2-(p_a+p_b-p_n)^2\big)\;\frac{|\boldsymbol{p}_n|^2\mathrm{d}|\boldsymbol{p}_n|\mathrm{d}\Omega_n}{2E_n}\,R_{n-1}\big(p_a+p_b-p_n\big)\\
&=\int \mathrm{d}M_{n-1}^2\,\frac{\delta\Big(|\boldsymbol{p}_n|-\sqrt{\lambda(s,M_{n-1}^2,m_n^2)}\Big/(2\sqrt{s})\Big)}{2\sqrt{s}\,|\boldsymbol{p}_n|/E_n}\;\frac{|\boldsymbol{p}_n|^2\mathrm{d}|\boldsymbol{p}_n|\mathrm{d}\Omega_n}{2E_n}\,R_{n-1}\big(p_a+p_b-p_n\big)\\
&=\int \mathrm{d}M_{n-1}^2\,\frac{\sqrt{\lambda(s,M_{n-1}^2,m_n^2)}}{8s}\,\mathrm{d}\Omega_n\,R_{n-1}\big(M_{n-1}^2\big).
\end{aligned}
$$

References

[1] E. Byckling and K. Kajantie. N-particle phase space in terms of invariant momentum transfers. *Nucl. Phys. B*, 9:568–576, 1969.

[2] Joshua Isaacson, Stefan Höche, Diego Lopez Gutierrez, and Noemi Rocco. Novel event generator for the automated simulation of neutrino scattering. *Phys. Rev. D*, 105(9):096006, 2022.