When we make the coordinate transformation $y = pa$, the boundaries of integration should be changed, as opposed to what we previously discussed (extending the boundaries so that momentum will still be within the grid).

$$\int_0^{p_{max}} dp = \frac{1}{a}\int_0^{p_{max}a} dy$$

So instead of putting the values of p in the collision integral, it should be $p * a$. Here p is the fixed momentum grid defined in the __main__ file.

The reason why I say this is because I noticed a difference between the combinations of p_1, p_2, p_3 for given p_0 considered in my code with physical coordinates and those in your code with comoving coordinates. There were a lot more combinations of the different momenta considered in the previous case. This shouldn't be so.

There's a part in the cpp code that checks whether $E_3 \geq m_3$ as follows in the current version:

$$\widetilde{E}_3 = aE_3 = \sqrt{y_0^2 + m_N^2 a^2} - \sqrt{y_1^2 + m_1^2 a^2} - \sqrt{y_2^2 + m_2^2 a^2} \geq m_3 * a$$

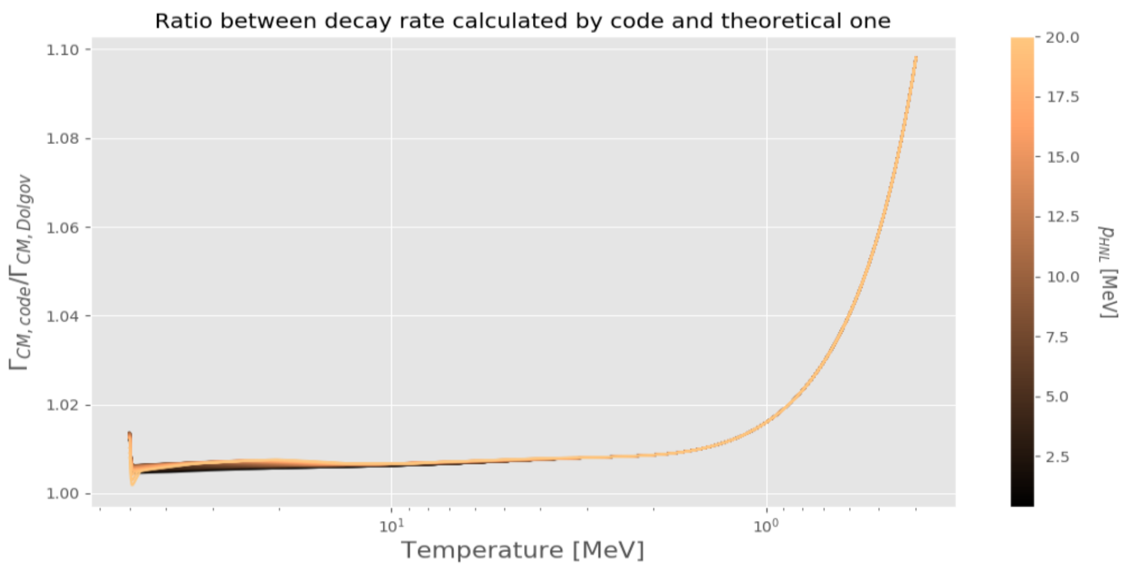
But for y_1 and y_2 you don't substitute $p_1 * a$ and $p_2 * a$. If you do the correct substitution, then you get:

$$\begin{aligned}
aE_3 &= \sqrt{y_0^2 + m_N^2 a^2} - \sqrt{p_1^2 a^2 + m_1^2 a^2} - \sqrt{p_2^2 a^2 + m_2^2 a^2} \geq m_3 * a \\
aE_3 &= a\sqrt{y_0^2/a^2 + m_N^2} - a\sqrt{p_1^2 + m_1^2} - a\sqrt{p_2^2 + m_2^2} \geq m_3 * a \\
E_3 &= \sqrt{y_0^2/a^2 + m_N^2} - \sqrt{p_1^2 + m_1^2} - \sqrt{p_2^2 + m_2^2} \geq m_3
\end{aligned}$$

And the last equation is indeed the correct check for a decaying particle in an expanding universe. Once I changed this in your code, I got the same combinations of p_1, p_2, p_3 as I did before when using physical coordinates.

So, in comoving coordinates the momenta of the decay products are not np.linspace(0, 20, 51) but np.linspace(0, 20 * a, 51). Since we start the simulation at $T = 50$ MeV, the scale factor starts at $1/50 = 0.02$. And this makes the difference, because in physical coordinates this means that we were considering HNLs with momenta $p = y/a =$ np.linspace(0, 20 / a, 51). With a $= 0.02$ this includes HNLs with physical momenta of 1000 MeV. This could explain all the wiggles in the plots before.

I made a test run with comoving momenta $p_0 = p_N =$ np.linspace(0, 0.01, 51), which correspond with maximum momentum of 0.5 MeV, and $p_1, p_2, p_3 =$ np.linspace(0, 20, 51). This gives the following plot:



I must note that not all plots look as fancy as this one.
Tweaking MAX_MOMENTUM of all particles and the order of Gaussian quadrature changes the shape of the plot in the beginning. But they are still very close to the theoretical value. Wiggles are gone too.

What do you think?