Building an Entropy based EOS Table

Alex Grannan

September 18, 2018

Abstract

In traditional EOS routines the electron positron EOS is based on table interpolation of the Helmholtz free energy $F(\rho, T)$ created from a biquintic Hermite polynomial interpolating function.

In this work, we want to create and entropy base interpolating function, $S(\rho, \epsilon_{tot})$ that uses the natural variables used in used by the hydrodynamics solvers.

$$d\epsilon = TdS + \frac{P}{\rho^2}d\rho \tag{1}$$

$$dS = \frac{1}{T}d\epsilon - \frac{P}{T\rho^2}d\rho \tag{2}$$

where $S = S(\rho, \epsilon)$, $T = T(\rho, \epsilon)$, and $P = P(\rho, \epsilon)$.

$$\left.\frac{\partial S}{\partial \epsilon}\right|_\rho = \frac{1}{T} \tag{3}$$

$$\left.\frac{\partial S}{\partial \rho}\right|_\epsilon = -\frac{P}{T\rho^2} \tag{4}$$

$$\left.\frac{\partial^2 S}{\partial \epsilon^2}\right|_\rho = -\frac{1}{T^2}\left.\frac{\partial T}{\partial \epsilon}\right|_\rho = -\frac{1}{T^2}\left(\left.\frac{\partial \epsilon}{\partial T}\right|_\rho\right)^{-1} \tag{5}$$

$$\left.\frac{\partial^2 S}{\partial \rho^2}\right|_\epsilon = \frac{2P}{T\rho^3} - \frac{1}{\rho^2}\left[-\frac{P}{T^2}\left.\frac{\partial T}{\partial \rho}\right|_\epsilon + \frac{1}{T}\left.\frac{\partial P}{\partial \rho}\right|_\epsilon\right] \tag{6}$$

$$\frac{\partial^2 S}{\partial \epsilon \partial \rho} = -\frac{1}{\rho^2}\left[-\frac{P}{T^2}\left.\frac{\partial T}{\partial \epsilon}\right|_\rho + \frac{1}{T}\left.\frac{\partial P}{\partial \epsilon}\right|_\rho\right] = \frac{\partial^2 S}{\partial \rho \partial \epsilon} = -\frac{1}{T^2}\left.\frac{\partial T}{\partial \rho}\right|_\epsilon = -\frac{1}{T^2}\left.\left(\frac{\partial \rho}{\partial T}\right)^{-1}\right|_\epsilon \tag{7}$$

1 Basic Equations

$$P_{tot} = P_{rad} + P_{ion} + P_{ele} + P_{pos} \tag{8}$$

$$\epsilon_{tot} = \epsilon_{rad} + \epsilon_{ion} + \epsilon_{ele} + \epsilon_{pos} \tag{9}$$

$$S_{tot} = S_{rad} + S_{ion} + S_{ele} + S_{pos} \tag{10}$$

1.1 Radiative Terms

$$P_{rad} = \frac{aT^4}{3} = \frac{a(m_e c^2)^4}{3k^4}\beta^4 \tag{11}$$

$$\epsilon_{rad} = \frac{3P_{rad}}{\rho} = \frac{a(m_e c^2)^4}{\rho k^4}\beta^4 \tag{12}$$

$$S^*_{rad} = \frac{P_{rad}/\rho + \epsilon_{rad}}{T} \tag{13}$$

where a is related to the Stephan-Boltzmann constant, $\sigma_B = ac/4$ and c is the speed of light.

1.2 Ion Terms

$$N_{ion} = \frac{N_A \rho}{\bar{A}} \tag{14}$$

$$P_{ion} = \frac{N_A k}{\bar{A}} \rho T = \frac{N_A \rho}{\bar{A}} \beta m_e c^2 \tag{15}$$

$$\epsilon_{ion} = \frac{3}{2} \frac{P_{ion}}{\rho} = \frac{3}{2} \frac{N_A}{\bar{A}} \beta m_e c^2 \tag{16}$$

$$S^*_{ion} = \frac{P_{ion}/\rho + \epsilon_{ion}}{T} + \frac{N_A k}{\bar{A}} log \left[\frac{N_A \rho}{\bar{A}^{5/2}} \left(\frac{N_A h^2}{2\pi k T} \right)^{3/2} \right] = \frac{N_A k}{\bar{A}} \left[\frac{5}{2} - \eta_{ion} \right] \tag{17}$$

where $N_A = 1/1amu$ is used in the Timmes EOS code and h is Planck's constant. Now for the derivatives of S_{ion}

$$\left. \frac{\partial S_{ion}}{\partial \rho} \right|_\epsilon = \frac{N_A k}{\bar{A}} \left. \frac{\partial \eta_{ion}}{\partial \rho} \right|_\epsilon \tag{18}$$

$$\left. \frac{\partial^2 S_{ion}}{\partial \rho^2} \right|_\epsilon = \frac{N_A k}{\bar{A}} \left. \frac{\partial^2 \eta_{ion}}{\partial \rho^2} \right|_\epsilon \tag{19}$$

1.3 Electron-Positron Terms

Both stellar EOS routines in FLASH are based on the formalism of a noninteracting Fermi gas for the electrons and positrons. The number density of free electrons N_{ele} and positrons N_{pos} in this formalism is given by

$$N^*_{ele} = \frac{8\pi\sqrt{2}}{h^3} m_e^3 c^3 \beta^{3/2} \left[F_{1/2}(\eta, \beta) + \beta F_{3/2}(\eta, \beta) \right] \tag{20}$$

$$N_{pos} = \frac{8\pi\sqrt{2}}{h^3} m_e^3 c^3 \beta^{3/2} \left[F_{1/2}(-\eta - 2\beta, \beta) + \beta F_{3/2}(-\eta - 2\beta, \beta) \right] \tag{21}$$

where m_e is the electron rest mass, the relativity parameter β is

$$\beta = kT/(m_e c^2) \tag{22}$$

and the normalized chemical potential energy μ of electrons is

$$\eta = \mu/kT, \tag{23}$$

and $F_k(\eta, \beta)$ is the Fermi-Dirac integral

$$F_k(\eta, \beta) = \int_0^\infty \frac{x^k (1 + 0.5\beta x)^{1/2} dx}{\exp(x - \eta) + 1} \tag{24}$$

The normalized chemical potential η in this formalism has the rest mass energy of the electrons subtracted out. This means that the positron chemical potential must have the rest mass terms appear explicitly, $\eta_{pos} = -\eta - 2/\beta$ as it does in (21).

For complete ionization, the number density of free electrons in the matter is

$$N_{ele,matter} = \frac{\bar{Z}}{\bar{A}} N^*_A \rho = \bar{Z} N_{ion}, \tag{25}$$

and charge neutrality requires

$$N_{ele,matter} = N_{ele} - N_{pos} \tag{26}$$

In order to calculate $\frac{\partial^2 S}{\partial \rho \partial \epsilon}$ from (55), the density can be written in terms of the temperature.

$$\rho = \frac{1}{N_A} \frac{\bar{A}}{\bar{Z}} \left[N_{ele} - N_{pos} \right] \tag{27}$$

Solving equation (26) determines the normalized chemical potential η, which was the only unknown. Such a solution fulfills the chemical potentials role as the Lagrange multiplier that was originally introduced to constrain

the distribution function to have the correct number of particles. Solving equation (26) in practice means using a one-dimensional root find method to obtain the root η.

Once η is known from the solution of equation (26), the pressure, specific internal energy, and entropy due to the free electrons and positrons are

$$P^*_{ele} = \frac{16\pi\sqrt{2}}{3h^3} m_e^4 c^5 \beta^{5/2} \left[F_{3/2}(\eta,\beta) + \frac{1}{2}\beta F_{5/2}(\eta,\beta) \right] \tag{28}$$

$$P^*_{pos} = \frac{16\pi\sqrt{2}}{3h^3} m_e^4 c^5 \beta^{5/2} \left[F_{3/2}(-\eta-2/\beta,\beta) + \frac{1}{2}\beta F_{5/2}(-\eta-2/\beta,\beta) \right] \tag{29}$$

$$\epsilon_{ele} = \frac{8\pi\sqrt{2}}{\rho h^3} m_e^4 c^5 \beta^{5/2} \left[F_{3/2}(\eta,\beta) + \beta F_{5/2}(\eta,\beta) \right] \tag{30}$$

$$\epsilon_{pos} = \frac{8\pi\sqrt{2}}{\rho h^3} m_e^4 c^5 \beta^{5/2} \left[F_{3/2}(-\eta-2/\beta,\beta) + \beta F_{5/2}(-\eta-2/\beta,\beta) \right] + \frac{2 m_e c^2 N_{pos}}{\rho} \tag{31}$$

$$S_{ele} = \frac{P_{ele}/\rho + \epsilon_{ele}}{T} - \frac{k\eta N_{ele}}{\rho} \tag{32}$$

$$S_{pos} = \frac{P_{pos}/\rho + \epsilon_{pos}}{T} + \frac{k\eta_{pos} N_{pos}}{\rho} \tag{33}$$

2 Solving the Fermi-Dirac Integrals

// The Fermi-Dirac integrals are solved following the model of Aparicio 1998 and coded in the Timmes EOS Fortran routine. From the Aparicio 1998 paper, the integral is found to be most accurate when broken into four separate integrals

$$F_k(\eta,\beta) = \int_0^\infty \frac{x^k (1+\beta x/2)^{1/2} dx}{\exp(x-\eta)+1} = \underbrace{\int_0^{S_1} f_1(x=z^2) dx}_{(1)} + \underbrace{\int_{S_1}^{S_2} f_2(x) dx}_{(2)} + \underbrace{\int_{S_2}^{S_3} f_2(x) dx}_{(3)} + \underbrace{\int_{S_3}^{\infty} f_2(x) dx}_{(4)} \tag{34}$$

For f_1 and f_2, the integrand of the generalized Fermi-Dirac integrand is rewritten based on two conditions:

$$f_1(z) = \begin{cases} \frac{2z^{2k+1}\sqrt{1+z^2\beta/2}}{\exp(z^2-\eta)+1}, & \text{if } (z-\eta) < 100; \\ 2z^{2k+1}\sqrt{1+z^2\beta/2}\exp(\eta-z^2), & \text{otherwise;} \end{cases} \tag{35}$$

$$f_2(x) = \begin{cases} \frac{x^k\sqrt{1+x\beta/2}}{\exp(x-\eta)+1}, & \text{if } (x-\eta) < 100; \\ x^k\sqrt{1+x\beta/2}\exp(\eta-x), & \text{otherwise;} \end{cases} \tag{36}$$

- Integrals (1), (2), and (3) use Gauss-Legendre (G-Le) quadrature for calculating the integrals
- Integral (4) uses Gauss-Laguerre (G-La) quadrature to calculate the integral

3 Calculating the Minimum Energy

A minimum energy for a completley degenerate gas is given in the limit that $T = 0$ and $\eta = \mu/kT \to \infty$ We start by first finding the Fermi momentum, p_F, by solving (25) in the above limit. As such, neglecting any contribution from positrons yields the equation:

$$\frac{\bar{Z}}{\bar{A}} N^*_A \rho = N_{ele} = \frac{2}{h^3}\int_0^{p_F} 4\pi p^2 dp = \frac{8\pi}{3h^3} p_F^3 \to p_F = \left(\frac{\bar{Z}}{\bar{A}} N^*_A \rho \frac{3h^3}{8\pi} \right)^{1/3} \tag{37}$$

The total energy over all possible states is rewritten from (30) as

$$\epsilon_{ele} = \frac{8\pi\sqrt{2}}{\rho h^3} m_e^4 c^5 \int_0^\infty \frac{(x\beta)^{3/2}(1+0.5x\beta)^{1/2}(1+x\beta)}{\exp(x-\eta)+1} \beta dx \tag{38}$$

Recall that in the above equation, $\beta = kT/mc^2$, $\eta = \mu/kT$, and $x = \epsilon_{kin}/kT$. With the inclusion of relativistic corrections the kinetic energy is given by

$$\epsilon_{kin} = mc^2 \left[\left[\left(\frac{p}{mc} \right)^2 + 1 \right]^{1/2} - 1 \right] = \epsilon_p - mc^2 \tag{39}$$

The minimum energy is calculated by integrating the above equation from 0 to x_F where the subscript F denotes the the Fermi energy and associated momentum. In this limit, $[\exp(x-\eta)+1]^{-1} \to 1$ so the integral may be rewritten as

$$\epsilon_{min} = \frac{8\pi\sqrt{2}}{\rho h^3} m_e^4 c^5 \int_0^{x_F} (x\beta)^{3/2}(1+0.5x\beta)^{1/2}(1+x\beta)\beta dx \tag{40}$$

We now make the substitution $\theta = x\beta = \epsilon_{kin}/mc^2$. The integral in (40) is then written as

$$\epsilon_{min} = \frac{8\pi\sqrt{2}}{\rho h^3} m_e^4 c^5 \int_0^{\theta_F} (\theta)^{3/2}(1+0.5\theta)^{1/2}(1+\theta) d\theta \tag{41}$$

NOTE: I have not been able to find an analytic solution to the above integral but I could determine it numerically. However we can take a look at some limiting cases for the integrand

$$g(\theta) = \theta^{3/2}(1+0.5\theta)^{1/2}(1+\theta) \tag{42}$$

$$\lim_{\theta\to 0} g(x) = \theta^{3/2} + 1.25\theta^{5/2} + 0.21875\theta^{7/2} - 0.0234374\theta^{9/2} + \mathcal{O}(\theta^{11/2}) \tag{43}$$

$$\lim_{\theta\to \infty} g(x) = 0.707107\theta^3 + 1.41421\theta^2 + 0.353553\theta + \mathcal{O}(\sqrt{1/\theta}) \tag{44}$$

The integrals in these limits can then be computed,

$$\int_0^{\theta_F} \lim_{\theta\to 0} g(x) dx = 0.4\theta_F^{5/2} + 0.357143\theta_F^{7/2} + 0.0486111\theta_F^{9/2} - 0.00426136\theta_F^{11/2} + \cdots \tag{45}$$

$$\int_0^{\theta_F} \lim_{\theta\to \infty} g(x) dx = 0.176777\theta_F^4 + 0.471403\theta_F^3 + 0.176777\theta_F^2 + \cdots \tag{46}$$

$$\epsilon_{min} = \frac{2}{h}\int_0^{p_F} \left(p^2c^2 + m_e^2c^4\right)^{1/2} 4\pi p^2 dp = \frac{m_e c^2}{\lambda_e^3}\chi(x) = \frac{m_e c^2}{8\pi^2\lambda_e^3}\left[x(1+x^2)^{1/2}(1+2x^2) - \ln\left[x+(1+x^2)^{1/2}\right]\right] \tag{47}$$

Thus an approximation for the minimum energy can be made using the above integrals and the prefactor from (40). These approximations describe the values obtained from Timmes EOS scheme and the current EOS scheme being written.

4 Creating Entropy-based EOS Table

$$\epsilon_3 = \epsilon_{ion} + \epsilon_{ele} + \epsilon_{pos} \tag{48}$$

$$\boxed{S_3 = S_{ion} + S_{ele} + S_{pos}} \tag{49}$$

$$P_3 = P_{ion} + P_{ele} + P_{pos} \tag{50}$$

$$\boxed{\left.\frac{\partial S_3}{\partial \epsilon_3}\right|_\rho = \frac{1}{T}} \tag{51}$$

$$\boxed{\left.\frac{\partial S_3}{\partial \rho}\right|_\epsilon = -\frac{P_3}{T\rho^2}} \tag{52}$$

$$\boxed{\left.\frac{\partial^2 S_3}{\partial \epsilon_3^2}\right|_\rho = -\frac{1}{T^2}\left.\frac{\partial T}{\partial \epsilon_3}\right|_\rho = -\frac{1}{T^2}\left(\left.\frac{\partial \epsilon_3}{\partial T}\right|_\rho\right)^{-1}} \tag{53}$$

$$\boxed{\left.\frac{\partial^2 S_3}{\partial \rho^2}\right|_\epsilon = \frac{2P_3}{T\rho^3} - \frac{1}{\rho^2}\left[-\frac{P_3}{T^2}\left(\left.\frac{\partial \rho}{\partial T}\right|_\epsilon\right)^{-1} + \frac{1}{T}\left.\frac{\partial P_3}{\partial \rho}\right|_\epsilon\right]} \tag{54}$$

$$\frac{\partial^2 S_3}{\partial \epsilon_3 \partial \rho} = \frac{P_3}{\rho^2 T^2}\left(\left.\frac{\partial \epsilon_3}{\partial T}\right|_\rho\right)^{-1} - \frac{1}{T\rho^2}\left.\frac{\partial P_3}{\partial \epsilon_3}\right|_\rho = \frac{P_3}{\rho^2 T^2}\left(\left.\frac{\partial \epsilon_3}{\partial T}\right|_\rho\right)^{-1} - \frac{2}{3}\frac{1}{\rho T} \tag{55}$$

$$\boxed{\frac{\partial^2 S_3}{\partial \rho \partial \epsilon_3} = -\frac{1}{T^2}\left.\frac{\partial T}{\partial \rho}\right|_\epsilon = -\frac{1}{T^2}\left(\left.\frac{\partial \rho}{\partial T}\right|_\epsilon\right)^{-1}} \tag{56}$$

5 Building EOS from Gaussian Process

$$\boxed{T = \left(\left.\frac{\partial S_3}{\partial \epsilon_3}\right|_\rho\right)^{-1}} \tag{57}$$

$$\boxed{P_3 = -\rho^2 T \left.\frac{\partial S_3}{\partial \rho}\right|_\epsilon} \tag{58}$$

6 Gaussian Process

In order to generate an interpolation of enetropy and its derivatives by gaussian process, we begin by assuming there exists a large four-dimensional lattice. For this section, the four dimensions are energy denoted by w, density denoted by x, Ye denoted by y, and $1/\bar{A}$ denoted by z. We assume that a single four dimensional cell or hypercube in this lattice is composed of N vertices or points where the entropy, its four first derivatives, and ten second derivatives are saved. The smallest hypercube has $N = 16$ points and we will assume below that we are interpolating somewhere inside a single hypercube, $c = 1$.

6.1 Setting up covariance matrix

Interpolation by gaussian process begins by building the covariance matrix. For this work we use a squared exponential covariance kernel, $\mathbf{k}$, that is constructed using a set of N points described by a four variable vector $\mathbf{x}_i = (w_i, x_i, y_i, z_i)$ and $\mathbf{x}_j = (w_j, x_j, y_j, z_j)$ where $i, j = 1 - N$. An $N \times N$ matrix, $\mathbf{k}$ is generate with each element given by

$$k(\mathbf{x}_i, \mathbf{x}_j) = k_{ij} = \sigma_f^2 \exp\left[-\frac{1}{2}\left[\frac{(w_i - w_j)^2}{\ell_{w,c}^2} + \frac{(x_i - x_j)^2}{\ell_{x,c}^2} + \frac{(y_i - y_j)^2}{\ell_{y,c}^2} + \frac{(z_i - z_j)^2}{\ell_{z,c}^2}\right]\right] \tag{59}$$

where the hyperparameters $\sigma_f = 1$ is the maximum variance and $\ell_{w,c}$, $\ell_{x,c}$, $\ell_{y,c}$, $\ell_{z,c}$ are the length scales in the four directions that are specific to each cell, c, and not to the points. Note that k_{ij} has a maximum value of 1 when $i = j$ for instance or when the length scales are very large. The minimum value is zero, as in when, for fixed length scales, the distance between points at i and j are very large for instance.

The first derivatives can be constructed as follows

$$\nabla_{\mathbf{x}_i}\mathbf{k} = \left[\frac{\partial \mathbf{k}}{\partial w_i}, \frac{\partial \mathbf{k}}{\partial x_i}, \frac{\partial \mathbf{k}}{\partial y_i}, \frac{\partial \mathbf{k}}{\partial z_i}\right] = -\left[\frac{(w_i - w_j)}{\ell_{w,c}^2}, \frac{(x_i - x_j)}{\ell_{x,c}^2}, \frac{(y_i - y_j)}{\ell_{y,c}^2}, \frac{(z_i - z_j)}{\ell_{z,c}^2}\right]\mathbf{k} \tag{60}$$

$$\nabla_{\mathbf{x}_j}\mathbf{k} = \left[\frac{\partial \mathbf{k}}{\partial w_j}, \frac{\partial \mathbf{k}}{\partial x_j}, \frac{\partial \mathbf{k}}{\partial y_j}, \frac{\partial \mathbf{k}}{\partial z_j}\right] = \left[\frac{(w_i - w_j)}{\ell_{w,c}^2}, \frac{(x_i - x_j)}{\ell_{x,c}^2}, \frac{(y_i - y_j)}{\ell_{y,c}^2}, \frac{(z_i - z_j)}{\ell_{z,c}^2}\right]\mathbf{k} \tag{61}$$

The other derivatives are found in a similar way. The joint covariance matrix includes the weights for the standard covariance as well as the the gradient and Laplacian terms:

$$\mathbf{K} = \begin{bmatrix}
\mathbf{k} & \frac{\partial \mathbf{k}}{\partial w_j} & \frac{\partial \mathbf{k}}{\partial x_j} & \frac{\partial \mathbf{k}}{\partial y_j} & \frac{\partial \mathbf{k}}{\partial z_j} & \frac{\partial^2 \mathbf{k}}{\partial w_j^2} & \frac{\partial^2 \mathbf{k}}{\partial w_j \partial x_j} & \cdots & \frac{\partial^2 \mathbf{k}}{\partial y_j \partial z_j} & \frac{\partial^2 \mathbf{k}}{\partial z_j^2} \\
\frac{\partial \mathbf{k}}{\partial w_i} & \frac{\partial^2 \mathbf{k}}{\partial w_i \partial w_j} & \frac{\partial^2 \mathbf{k}}{\partial w_i \partial x_j} & \frac{\partial^2 \mathbf{k}}{\partial w_i \partial y_j} & \frac{\partial^2 \mathbf{k}}{\partial w_i \partial z_j} & \frac{\partial^3 \mathbf{k}}{\partial w_i \partial w_j^2} & \frac{\partial^3 \mathbf{k}}{\partial w_i \partial w_j \partial x_j} & \cdots & \frac{\partial^3 \mathbf{k}}{\partial w_i \partial y_j \partial z_j} & \frac{\partial^3 \mathbf{k}}{\partial w_i \partial z_j^2} \\
\frac{\partial \mathbf{k}}{\partial x_i} & \frac{\partial^2 \mathbf{k}}{\partial x_i \partial w_j} & \frac{\partial^2 \mathbf{k}}{\partial x_i \partial x_j} & \frac{\partial^2 \mathbf{k}}{\partial x_i \partial y_j} & \frac{\partial^2 \mathbf{k}}{\partial x_i \partial z_j} & \frac{\partial^3 \mathbf{k}}{\partial x_i \partial w_j^2} & \frac{\partial^3 \mathbf{k}}{\partial x_i \partial w_j \partial x_j} & \cdots & \frac{\partial^3 \mathbf{k}}{\partial x_i \partial y_j \partial z_j} & \frac{\partial^3 \mathbf{k}}{\partial x_i \partial z_j^2} \\
\frac{\partial \mathbf{k}}{\partial y_i} & \frac{\partial^2 \mathbf{k}}{\partial y_i \partial w_j} & \frac{\partial^2 \mathbf{k}}{\partial y_i \partial x_j} & \frac{\partial^2 \mathbf{k}}{\partial y_i \partial y_j} & \frac{\partial^2 \mathbf{k}}{\partial y_i \partial z_j} & \frac{\partial^3 \mathbf{k}}{\partial y_i \partial w_j^2} & \frac{\partial^3 \mathbf{k}}{\partial y_i \partial w_j \partial x'} & \cdots & \frac{\partial^3 \mathbf{k}}{\partial y_i \partial y_j \partial z_j} & \frac{\partial^3 \mathbf{k}}{\partial y_i \partial z_j^2} \\
\frac{\partial \mathbf{k}}{\partial z_i} & \frac{\partial^2 \mathbf{k}}{\partial z_i \partial w_j} & \frac{\partial^2 \mathbf{k}}{\partial z_i \partial x_j} & \frac{\partial^2 \mathbf{k}}{\partial z_i \partial y_j} & \frac{\partial^2 \mathbf{k}}{\partial z_i \partial z_j} & \frac{\partial^3 \mathbf{k}}{\partial z_i \partial w_j^2} & \frac{\partial^3 \mathbf{k}}{\partial z_i \partial w_j \partial x_j} & \cdots & \frac{\partial^3 \mathbf{k}}{\partial z_i \partial y_j \partial z_j} & \frac{\partial^3 \mathbf{k}}{\partial z_i \partial z_j^2} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\frac{\partial^2 \mathbf{k}}{\partial y_i \partial z_i} & \frac{\partial^3 \mathbf{k}}{\partial y_i \partial z_i \partial w_j} & \frac{\partial^3 \mathbf{k}}{\partial y_i \partial z_i \partial x_j} & \frac{\partial^3 k}{\partial y_i \partial z_i \partial y_j} & \frac{\partial^3 \mathbf{k}}{\partial y_i \partial z_i \partial z_j} & \frac{\partial^4 \mathbf{k}}{\partial y_i \partial z_i \partial w_j^2} & \frac{\partial^4 \mathbf{k}}{\partial y_i \partial z_i \partial w_j \partial x_j} & \cdots & \frac{\partial^4 \mathbf{k}}{\partial y_i \partial z_i \partial y_j \partial z_j} & \frac{\partial^4 \mathbf{k}}{\partial y_i \partial z_i \partial z_j^2} \\
\frac{\partial^2 \mathbf{k}}{\partial z_i^2} & \frac{\partial^3 \mathbf{k}}{\partial z_i^2 \partial w_j} & \frac{\partial^3 \mathbf{k}}{\partial z_i^2 \partial x_j} & \frac{\partial^3 \mathbf{k}}{\partial z_i^2 \partial y_j} & \frac{\partial^3 \mathbf{k}}{\partial z_i^2 \partial z_j} & \frac{\partial^4 \mathbf{k}}{\partial z_i^2 \partial w_j^2} & \frac{\partial^4 \mathbf{k}}{\partial z_i^2 \partial w_j \partial x_j} & \cdots & \frac{\partial^4 \mathbf{k}}{\partial z_i^2 \partial y_j \partial z_j} & \frac{\partial^4 \mathbf{k}}{\partial z_i^2 \partial z_j^2}
\end{bmatrix} + \sigma_n \mathbf{I} \tag{62}$$

To be clear, the joint covariance is composed of the known or training points. The largest, $\mathbf{K}$ will be is $15N \times 15N$ if all derivatives are included. The point for interpolation is given by $\mathbf{x}^* = (w^*, x^*, y^*, z^*)$. A covariance matrix, $\mathbf{k}^*$ can be made between the training points and the interpolating variables, each element is definfed by

$$k(\mathbf{x}^*, \mathbf{x}_j) = k_j^* = \sigma_f^2 \exp\left[-\frac{1}{2}\left[\frac{(w^* - w_j)^2}{\ell_{w,c}^2} + \frac{(x^* - x_j)^2}{\ell_{x,c}^2} + \frac{(y^* - y_j)^2}{\ell_{y,c}^2} + \frac{(z^* - z_j)^2}{\ell_{z,c}^2}\right]\right] \tag{63}$$

By using a single interpolating point, $\mathbf{k}^*$ is a $1 \times N$ vector. The full matrix is then given by

$$\mathbf{K}^* = \begin{bmatrix}
\mathbf{k}^* & \frac{\partial \mathbf{k}^*}{\partial w_j} & \frac{\partial \mathbf{k}^*}{\partial x_j} & \frac{\partial \mathbf{k}^*}{\partial y_j} & \frac{\partial \mathbf{k}^*}{\partial z_j} & \frac{\partial^2 \mathbf{k}^*}{\partial w_j^2} & \frac{\partial^2 \mathbf{k}^*}{\partial w_j \partial x_j} & \cdots & \frac{\partial^2 \mathbf{k}^*}{\partial y_j \partial z_j} & \frac{\partial^2 \mathbf{k}^*}{\partial z_j^2} \\
\frac{\partial \mathbf{k}^*}{\partial w^*} & \frac{\partial^2 \mathbf{k}^*}{\partial w^* \partial w_j} & \frac{\partial^2 \mathbf{k}^*}{\partial w * \partial x_j} & \frac{\partial^2 \mathbf{k}^*}{\partial w^* \partial y_j} & \frac{\partial^2 \mathbf{k}^*}{\partial w^* \partial z_j} & \frac{\partial^3 \mathbf{k}^*}{\partial w^* \partial w_j^2} & \frac{\partial^3 \mathbf{k}^*}{\partial w^* \partial w_j \partial x_j} & \cdots & \frac{\partial^3 \mathbf{k}^*}{\partial w^* \partial y_j \partial z_j} & \frac{\partial^3 \mathbf{k}^*}{\partial w^* \partial z_j^2} \\
\frac{\partial \mathbf{k}^*}{\partial x^*} & \frac{\partial^2 \mathbf{k}^*}{\partial x^* \partial w_j} & \frac{\partial^2 \mathbf{k}^*}{\partial x^* \partial x_j} & \frac{\partial^2 \mathbf{k}^*}{\partial x^* \partial y_j} & \frac{\partial^2 \mathbf{k}^*}{\partial x^* \partial z_j} & \frac{\partial^3 \mathbf{k}^*}{\partial x^* \partial w_j^2} & \frac{\partial^3 \mathbf{k}^*}{\partial x^* \partial w_j \partial x_j} & \cdots & \frac{\partial^3 \mathbf{k}^*}{\partial x^* \partial y_j \partial z_j} & \frac{\partial^3 \mathbf{k}^*}{\partial x^* \partial z_j^2} \\
\frac{\partial \mathbf{k}^*}{\partial y^*} & \frac{\partial^2 \mathbf{k}^*}{\partial y^* \partial w_j} & \frac{\partial^2 \mathbf{k}^*}{\partial y^* \partial x_j} & \frac{\partial^2 \mathbf{k}^*}{\partial y^* \partial y_j} & \frac{\partial^2 \mathbf{k}^*}{\partial y^* \partial z_j} & \frac{\partial^3 \mathbf{k}^*}{\partial y^* \partial w_j^2} & \frac{\partial^3 \mathbf{k}^*}{\partial y^* \partial w_j \partial x'} & \cdots & \frac{\partial^3 \mathbf{k}^*}{\partial y^* \partial y_j \partial z_j} & \frac{\partial^3 \mathbf{k}^*}{\partial y^* \partial z_j^2} \\
\frac{\partial \mathbf{k}^*}{\partial z^*} & \frac{\partial^2 \mathbf{k}^*}{\partial z^* \partial w_j} & \frac{\partial^2 \mathbf{k}^*}{\partial z^* \partial x_j} & \frac{\partial^2 \mathbf{k}^*}{\partial z^* \partial y_j} & \frac{\partial^2 \mathbf{k}^*}{\partial z^* \partial z_j} & \frac{\partial^3 \mathbf{k}^*}{\partial z^* \partial w_j^2} & \frac{\partial^3 \mathbf{k}^*}{\partial z^* \partial w_j \partial x_j} & \cdots & \frac{\partial^3 \mathbf{k}^*}{\partial z^* \partial y_j \partial z_j} & \frac{\partial^3 \mathbf{k}^*}{\partial z^* \partial z_j^2} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\frac{\partial^2 \mathbf{k}^*}{\partial y^* \partial z^*} & \frac{\partial^3 \mathbf{k}^*}{\partial y^* \partial z^* \partial w_j} & \frac{\partial^3 \mathbf{k}^*}{\partial y^* \partial z^* \partial x_j} & \frac{\partial^3 k^*}{\partial y^* \partial z^* \partial y_j} & \frac{\partial^3 \mathbf{k}}{\partial y^* \partial z^* \partial z_j} & \frac{\partial^4 \mathbf{k}^*}{\partial y^* \partial z^* \partial w_j^2} & \frac{\partial^4 \mathbf{k}^*}{\partial y^* \partial z^* \partial w_j \partial x_j} & \cdots & \frac{\partial^4 \mathbf{k}^*}{\partial y^* \partial z^* \partial y_j \partial z_j} & \frac{\partial^4 \mathbf{k}^*}{\partial y^* \partial z^* \partial z_j^2} \\
\frac{\partial^2 \mathbf{k}^*}{\partial z^{*2}} & \frac{\partial^3 \mathbf{k}^*}{\partial z^{*2} \partial w_j} & \frac{\partial^3 \mathbf{k}^*}{\partial z^{*2} \partial x_j} & \frac{\partial^3 \mathbf{k}^*}{\partial z^{*2} \partial y_j} & \frac{\partial^3 \mathbf{k}^*}{\partial z^{*2} \partial z_j} & \frac{\partial^4 \mathbf{k}^*}{\partial z_i^2 \partial w_j^2} & \frac{\partial^4 \mathbf{k}^*}{\partial z^{*2} \partial w_j \partial x_j} & \cdots & \frac{\partial^4 \mathbf{k}^*}{\partial z^{2*} \partial y_j \partial z_j} & \frac{\partial^4 \mathbf{k}^*}{\partial z^{*2} \partial z_j^2}
\end{bmatrix} \tag{64}$$

The size of $\mathbf{K}^*$ is $15 \times 15N$ if first and second derivatives are included.

6.2 Finding the optimum hyperparameters

We compute the optimum values for the four length scales $[\ell_{w,c}, \ell_{w,c}, \ell_{y,c}, \ell_{z,c}]$ by minimizing the log marginal likelihood function determined trhought the following steps

1. $L = \text{cholesky}(\mathbf{K})$: The cholesky decomposition is done using numpy in python, np.linalg.cholesky(K)
2. $\alpha = L^\top \setminus (L \setminus [\mathbf{f}, \nabla \mathbf{f}, \nabla^2 \mathbf{f}])$: This factor is found using scipy in python, scipy.linalg.cho_factor(K,f)
3. $\log p([\mathbf{f}, \nabla \mathbf{f}, \nabla^2 \mathbf{f}]|x) = -0.5 \left([\mathbf{f}, \nabla \mathbf{f}, \nabla^2 \mathbf{f}]^\top \alpha + 2 \sum_i \log L_{ii} + n/2 \log(2\pi) \right)$: log marginal likelihood function where n is the size of the matrix

In order to find the length scales that minimize the log marginal likelihood function, a Nelder-Mead method in python is used.

6.3 Computing the posterior

Given a new set of variables, $\mathbf{x}^*$, the posterior or interpolated value, $\mathbf{f}^* = f(\mathbf{x}^*)$, can be determined and its posterior mean and covariance is given by

$$[\mathbf{f}, \nabla \mathbf{f}, \nabla^2 \mathbf{f}](\mathbf{x}^*) = \mathbf{K}^* \mathbf{K}^{-1} [\mathbf{f}, \nabla \mathbf{f}, \nabla^2 \mathbf{f}]^\top \tag{65}$$

$$\sigma^2(\mathbf{x}^*) = k(\mathbf{x}^*, \mathbf{x}^*) - (\bar{\mathbf{k}}^*)^\top \mathbf{K}^{-1} \bar{\mathbf{k}}^* \tag{66}$$

So if we are only considering interpolating for entropy, S and the first two derivatives for energy, E, and density ρ then

$$\left[S^*, \frac{\partial S^*}{\partial E}, \frac{\partial S^*}{\partial \rho} \right] = \mathbf{K}^* \mathbf{K}^{-1} \left[S_{1-N}, \frac{\partial S}{\partial E}_{1-N}, \frac{\partial S}{\partial \rho}_{1-N} \right]^\top \tag{67}$$

where, for this problem, $\mathbf{K}^*$ is a $3 \times 3N$ matrix and $\mathbf{K}^{-1}$ is a $3N \times 3N$ matrix.