JOINT INSTITUTE OF NUCLEAR RESEARCH

Laboratory of Information Technologies

FINAL REPORT OF SUMMER STUDENT PROGRAM

Algorithm and program for solving systems of nonlinear equations describing the phenomenological models of a mixed phase of cold dense nuclear matter

Supervisor:
Alexander Ayriyan

Student:
Olga Korneeva

Period of participation:
4 July-28 Aug 2017

Dubna, 2017

Contents

Introduction and Motivation

Usually modeling of possible hadron-quark phase transition are made with use of the so-called Maxwell construction, where the two phases are assumed to be separated. However due to surface tension effects in the mixed phase with structure (pasta) could be thermodynamically preferred [1–5]. Thus a simple model of such a mixed phase equation of state parametrized by impact of structures in mixed phase of the pressure ΔP will be very useful for investigations of compact stars (e.g. [6]). Also, we need to have the program in order to systematically study the effect of possible mixed state of matter on the structure of neutron stars.

The model

Let us suppose that the hadroic and quark density phases are given with the thermodynamical potentials $P_H(\mu)$ and $P_Q(\mu)$ correspondingly (T = 0 case relevant for the NS modeling). We suppose that it is possible to have a phase transition between these two phases on the μ_c when one ignores any effect of the pasta mixing (Maxwell construction). So we have

$$P_Q(\mu_c) = P_H(\mu_c)$$

Now we can modify this situation assuming that close to the phase transition point the Equation of State (EoS) of both phases are changing in due to the effects of interaction of the phase there is an additional contribution in the pressure at μ_c (ΔP a constant value characterizing the transition) (see fig.1)

$$P_M(\mu_c) = P_H(\mu_c) + \Delta P$$

Using this ansatz one can assume that the pressure of mixed phase $P_M(\mu)$ has the following form

$$P_M(\mu) = \alpha(\mu - \mu_c)^p + \beta(\mu - \mu_c)^q + P_c + \Delta P$$

where $P_c = P_{H_c} = P_{Q_c}, p$ and q for instance are non-negative integers.
In this research we focused on function model 4 degrees of the mixed phase (e.g. $p = 4$ and $q = 2$)

The transition from H-phase to M-phase happens smoothly without a jump in the density $n(\mu) = dP(\mu)/d\mu$. Thus we have new unknowns μ_{cH} for transition from H-phase to M-phase and correspondingly μ_{cQ} for the transition from M-phase to Q-phase. So we have four unknown including the coffcient α and β. The transition conditions are

$$\begin{aligned} P_M(\mu_{cH}) &= P_H(\mu_{cH}) \\ P_M(\mu_{cQ}) &= P_Q(\mu_{cQ}) \\ n_M(\mu_{cH}) &= n_H(\mu_{cH}) \\ n_M(\mu_{cQ}) &= n_Q(\mu_{cQ}) \end{aligned}$$

Solving the equation for densities, one can find the values for critical chemical potentials.

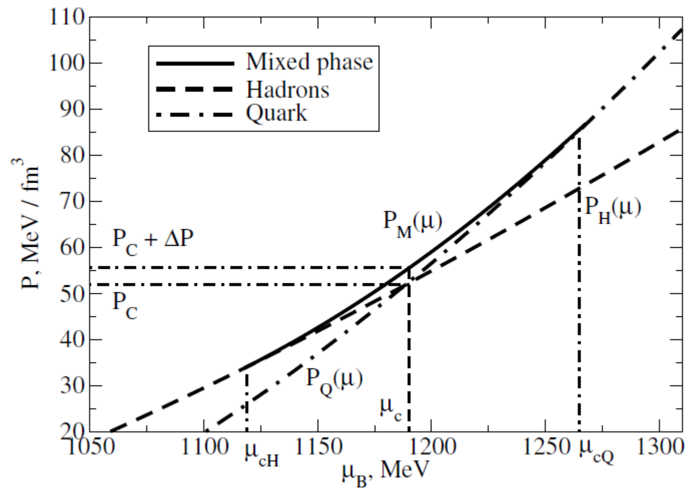

Figure 1: The model of the equation of state with mixed phase based on Maxwell construction.

Formulation of the problem

The purpose of my work is to develop an algorithm and a C ++ program for finding the coefficients of the equation

$$P(x) = \alpha(x - x_c)^4 + \beta(x - x_c)^2 + \gamma$$

describing the phenomenological models of a mixed phase of cold dense nuclear matter

There are initially 2 increasing functions $R(x)$ and $Q(x)$ and $Q(x)$ increases faster than $R(x)$. It is also known that these two functions intersect at some point xc.

$$R(x_c) = Q(x_c)$$

The value of the function $P(x)$ at the point xc

$$P(x_c) = Q(x_c) + \Delta P$$

where ΔP is the initial value.

To find the coefficients, it is necessary to solve a system of nonlinear equations:

$$\begin{cases} \alpha(x_l - x_c)^4 + \beta(x_l - x_c)^2 + \gamma &= f_1(x_l) \\ \alpha(x_r - x_c)^4 + \beta(x_r - x_c)^2 + \gamma &= f_1(x_r) \\ 4\alpha(x_r - x_c)^3 + 2\beta(x_r - x_c) &= f_2'(x_r) \\ 4\alpha(x_l - x_c)^3 + 2\beta(x_l - x_c) &= f_1'(x_l) \end{cases}$$

Algorithm for solving

It is required to develop an algorithm and implement it in C / C ++ for inclusion in the baYes software package (https://gitlab-hybrilit.jinr.ru/nmeos/cseos)

To find the solution of this system, the Newton method is used. The use of the Newton method implies the differentiability of functions $F_1(x), F_2(x), \ldots, F_n(x)$

$$\begin{cases} F_1 &=& \alpha(x_l - x_c)^4 + \beta(x_l - x_c)^2 + \gamma - f_1(x_l) \\ F_2 &=& \alpha(x_r - x_c)^4 + \beta(x_r - x_c)^2 + \gamma - f_1(x_r) \\ F_3 &=& 4\alpha(x_r - x_c)^3 + 2\beta(x_r - x_c) - f_2'(x_r) \\ F_4 &=& 4\alpha(x_l - x_c)^3 + 2\beta(x_l - x_c) - f_1'(x_l) \end{cases}$$

and the nonsingularity of the Jacobi matrix ($detJ(x_k) \neq 0$)

$$J(x) = \begin{bmatrix} \frac{\partial F_1(x)}{\partial x_1} \frac{\partial F_1(x)}{\partial x_2} \cdots \frac{\partial F_1(x)}{\partial x_n} \\ \frac{\partial F_2(x)}{\partial x_1} \frac{\partial F_2(x)}{\partial x_2} \cdots \frac{\partial F_2(x)}{\partial x_n} \\ \cdots \\ \frac{\partial F_n(x)}{\partial x_1} \frac{\partial F_n(x)}{\partial x_2} \cdots \frac{\partial F_n(x)}{\partial x_n} \end{bmatrix}$$

It is also necessary to know the initial approximation for the variables. For points x_l and x_r , the initial approximation ca be taken to be the point of intersection of x_c , but we do not know anything about α and β, therefore we know nothing about their initial approximation. As a result, we can apply the Newton method only for a system of two equations depending on x_l and x_r In order to bring the original system to a system of two equations, we express the α and β of their two equations and substitute their values in the remaining equations.

$$\alpha = -\frac{(x_c - x_r)f_1'(x_l) + (x_l - x_c)f_2'(x_r)}{4(x_c - x_l)(x_c - x_r)(x_l - x_r)(x_l + x_r - 2x_c)}$$

$$\beta = \frac{f_2'(x_r)(x_l - x_c)^3 - f_1'(x_l)(x_r - x_c)^3}{2(x_r - x_c)(x_l - x_c)((x_l - x_c)^2 - (x_r - x_c)^2)}$$

When substituting, we obtain a system of two equations

$$\begin{cases} F_1(x_l, x_r) = 0 \\ F_2(x_l, x_r) = 0 \end{cases}$$

when

$$F_1(x_l, x_r) = \gamma - f_1(x_l) - \frac{(x_l - x_c)^4((x_c - x_r)f_1'(x_l) + (x_l - x_c)f_2'(x_r))}{4(x_c - x_l)(x_c - x_r)(x_l - x_r)(x_l + x_r - 2x_c)} + \frac{(x_l - x_c)(-(x_r - x_c)^3 f_1'(x_l) + (x_l - x_c)^3 f_2'(x_r))}{2(x_r - x_c)((x_l - x_c)^2 - (x_r - x_c)^2)}$$

$$F_2(x_l,x_r) = \gamma - f_2(x_r) - \frac{(x_r-x_c)^4((x_c-x_r)f_1'(x_l)+(x_l-x_c)f_2'(x_r))}{4(x_c-x_l)(x_c-x_r)(x_l-x_r)(x_l+x_r-2x_c)} + \frac{(x_r-x_c)(-(x_r-x_c)^3 f_1'(x_l)+(x_l-x_c)^3 f_2'(x_r))}{2(x_l-x_c)((x_l-x_c)^2-(x_r-x_c)^2)}$$

The Jacobi matrix of this system of equations is as follows

$$J(x) = \begin{bmatrix} \frac{\partial F_1(x)}{\partial x_l} & \frac{\partial F_1(x)}{\partial x_r} \\ \frac{\partial F_2(x)}{\partial x_l} & \frac{\partial F_2(x)}{\partial x_r} \end{bmatrix}$$

when

$$F_1{}'_{x_l} = ((x_c^2 - x_l^2 + 2x_c(x_l-2x_r) + 2x_r^2)((x_c-x_r)(2x_c^2+3x_l^2-x_r^2+2x_c(-3x_l+x_r))f_1'(x_l) - 2(x_c-x_l)^3 f_2'(x_r) - (x_c-x_l)(x_c-x_r)(2x_c-x_l-x_r)(x_l-x_r)f_1{}''(x_l)))/ (4(x_c-x_r)(x_l-x_r)^2(x_l+x_r-2x_c)^2)$$

$$F_1{}'_{x_r} = ((x_c-x_l)^3(-2(x_c-x_r)^3 f_1'(x_l) + (x_c-x_l)((2x_c^2+2x_cx_l-x_l^2-6x_cx_r+3x_r^2)f_2'(x_r) + (x_c-x_r)(2x_c-x_l-x_r)(x_l-x_r)f_2{}''(x_r))))/(4(x_c-x_r)^2(x_l-x_r)^2(-2x_c+x_l+x_r)^2)$$

$$F_2{}'_{x_l} = ((x_c-x_r)^3((x_c-x_r)(2x_c^2+3x_l^2-x_r^2+2x_c(x_r-3x_l))f_1'(x_l) - 2(x_c-x_l)^3 f_2'(x_r) - (x_c-x_l)(x_c-x_r)(2x_c-x_l-x_r)(x_l-x_r)f_1{}''(x_l)))/(4(x_c-x_l)^2(x_l-x_r)^2(x_l+x_r-2x_c)^2)$$

$$F_2{}'_{x_r} = ((x_c^2+2x_l^2-x_r^2+2x_c(x_r-2x_l))(-2(x_c-x_r)^3 f_1'(x_l) + (x_c-x_l)((2x_c^2+2x_cx_l - x_l^2 - 6x_cx_r + 3x_r^2)f_2'(x_r) + (x_c-x_r)(2x_C-x_l-x_r)(x_l-x_r)f_2{}''(x_r)))) /(4(x_c-x_l)()x_l-x_r)^2(x_l+x_r-2x_c)^2$$

The initial approximation for x_l and x_r can take points around the point of intersection of lines

If it is determined the initial value ,the iterative process of finding the solution of a system Newton's method can be represented in the form

$$\begin{cases} x_1^{(k+1)} = x_1^{(k)} + \Delta x_1^{(k)} \\ x_2^{(k+1)} = x_2^{(k)} + \Delta x_2^{(k)} \\ \qquad \cdots \\ x_n^{(k+1)} = x_n^{(k)} + \Delta x_n^{(k)} \end{cases}$$

where the increments are determined from the solution of a system of linear algebraic equations, all the coefficients of which are expressed in terms of known previous approximation

$$\begin{cases} f_1(x^{(k)}) + \dfrac{\partial f_1(x^{(k)})}{\partial x_1}\Delta x_1^{(k)} + \dfrac{\partial f_1(x^{(k)})}{\partial x_2}\Delta x_2^{(k)} + \cdots + \dfrac{\partial f_1(x^{(k)})}{\partial x_n}\Delta x_n^{(k)} = 0 \\ f_2(x^{(k)}) + \dfrac{\partial f_2(x^{(k)})}{\partial x_1}\Delta x_1^{(k)} + \dfrac{\partial f_2(x^{(k)})}{\partial x_2}\Delta x_2^{(k)} + \cdots + \dfrac{\partial f_2(x^{(k)})}{\partial x_n}\Delta x_n^{(k)} = 0 \\ \cdots \\ f_n(x^{(k)}) + \dfrac{\partial f_n(x^{(k)})}{\partial x_1}\Delta x_1^{(k)} + \dfrac{\partial f_n(x^{(k)})}{\partial x_2}\Delta x_2^{(k)} + \cdots + \dfrac{\partial f_n(x^{(k)})}{\partial x_n}\Delta x_n^{(k)} = 0 \end{cases}$$

Finding solutions of system of linear equations is performed by Kramer General view of the solution

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \cdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}$$

$$\Delta x_1^k = \frac{-f_1(x^k){f_2}'x_r(x^k) + f_2(x^k){f_1}'x_r(x^k)}{{f_1}'_{x_l}(x^k){f_2}'_{x_r}(x^k) - {f_2}'x_l(x^k){f_1}'x_r(x^k)}$$

$$\Delta x_2^k = \frac{{f_1}'_{x_l}(x^k)(-f_2(x^k)) + f_1(x^k){f_2}'x_l}{{f_1}'_{x_l}(x^k){f_2}'_{x_r}(x^k) - {f_2}'x_l(x^k){f_1}'x_r(x^k)}$$

$$\begin{cases} x_1^{(k+1)} = x_1^{(k)} + \Delta x_1^{(k)} \\ x_2^{(k+1)} = x_2^{(k)} + \Delta x_2^{(k)} \end{cases}$$

As a condition of graduation from iterations commonly used criterion

$$|x^{(k+1)} - x^{(k)}| \leq \varepsilon$$

Conclusion

In the end, a program was written to find the coefficients of the given function. The program is written in both C++ and Wolfram mathematica. C++ program is planned to be integrated to the baYes software (https://gitlab-hlit.jinr.ru/nmeos/cseos) for statistical investigation of modern nuclear EoS using observational constants.

The solutions obtained by the program depend on the accuracy of what snicket error. The algorithm solving this problem does not depend on the initial given functions f_1 and f_2. Importantly, the program was able to get the value of these functions at a certain point, and the value of the derivative functions at this point.

Acknowledgments

I would like to express my gratitude to my scientific adviser, the scientific collaborator of the JINR, Alexander Ayriyan, who not only gave me an excellent opportunity to undergo an internship at JINR, but also provided support throughout the project.
I also would like to thank Hovik Grigoryan for fruitful discussions. Let me note that the considered phase transition construction mimicking mixed phase EoS is named Grigorian construction.

References

[1] D. N. Voskresensky et al., Nuclear Physics A 723, pp. 291–339 (2003), pp. 291–339
[2] T. Tatsumi et al., Nucl. Phys. A 718, pp. 359–362 (2003)
[3] T. Endo et al., Nucl. Phys. A 749, pp. 333–336 (2005)
[4] T. Endo et al., Prog. of Theor. Phys. 115(2), pp. 337–353 (2006)
[5] T. Tatsumi et al., arXiv:1107.0804 [nucl-th] (2011)
[6] T. Noda et al., Astrophys. J. 765:1 (2013)