Mapana Journal of Sciences
2019, Vol. 18, No. 1, 1-20
ISSN 0975-3303 | https://doi.org/10.12723/mjs.48.1

On the Adaptive Quadrature of Fermi-Dirac Functions and their Derivatives

Mandyam N Anandaram*

Abstract

In this paper, using the Python SciPy module "quad", a fast auto-adaptive quadrature solver based on the pre-compiled QUADPACK Fortran package, computational research is undertaken to accurately integrate the generalised Fermi-Dirac function and all its partial derivatives up to the third order. The numerical results obtained with quad method when combined with optimised break points achieve an excellent accuracy comparable to that obtained by other publications using fixed-order quadratures.

Keywords: Fermi-Dirac Integrals, Partial Derivatives, QUADPACK Adaptive Quadrature, Optimised Break Points

1. Introduction

Many calculations done in astrophysics and other fields require expressions for the various thermodynamic properties of partially degenerated electrons and other fermions in hot dense stellar plasmas. These properties are expressed in terms of Fermi-Dirac integrals (FDI) and their partial derivatives up to the third order. The non-relativistic FDI as a one-parameter integral of order $\boldsymbol{k}$ (aka the Fermi integral) is defined by,

$$F_k(\eta) = \int_0^\infty x^k dx/\big(1 + exp(x - \eta)\big), k > -1 \quad (1.1)$$

* Bangalore University, Bangalore, India; mnanandaram@gmail.com

Here, the degeneracy parameter $\boldsymbol{\eta \equiv \mu/k_B T}$ is the normalised chemical potential energy $\boldsymbol{\mu}$, $\boldsymbol{k_B}$ is the Boltzmann constant, **T** is the Kelvin temperature and $\boldsymbol{x}$ is the kinetic energy in the same units. The generalised two parameter FDI (GFDI) is defined by

$$F_k(\eta, \beta) = \int_0^\infty x^k \sqrt{(1+ \ \beta x/2)}\, dx/\big(1 + exp(x-\eta)\big), \qquad (1.2)$$

Here, the second argument is the relativity parameter $\boldsymbol{\beta \equiv k_B T/mc^2}$ which denotes thermal energy as a fraction of the rest mass energy of the fermion of interest. It may be seen that when $\boldsymbol{\beta = 0}$ this expression reduces to the non-relativistic expression (1.1).

The FDI of half-integral orders with $k = -0.5, 0.5, 1.5$ and so on finds applications in astrophysics. For instance, the number density of non-relativistic degenerate electron gas and its pressure are given, in the usual notation, respectively by

$$n_e = 4\pi(2m_e k_B T/h^2)^{3/2} F_{0.5}(\eta) \qquad (1.3)$$

$$P_e = (2/3) n_e k_B T\, F_{1.5}(\eta) \qquad (1.4)$$

In the collective opacity calculations, $F_{-0.5}$ is required. In the theory of partially degenerate electronic heat conductivity, FDIs of integer orders like F_2, F_3 and F_4 are required. In many professional stellar structures, tables of such FDIs which can be quickly interpolated are used. Older studies [2] used an approximation accurate to within ***0.02%*** for $\eta < 30$ and is given by,

$$F_{1.5}(\eta) = 1.5F_{0.5}(\eta)\big(1 + 0.1938F_{0.5}(\eta)\big)/\big(1 + 0.12398F_{0.5}(\eta)\big) \qquad (1.5)$$

The underlying issue, however, is accurate computation of these FDIs and GFDIs since the integrand varies as x^k when $x \to 0$ and decays as $\exp(-x)$ when $x \to \infty$. Over the years, two methods have become available. The less accurate method is the Cloutman method [5] which uses direct Simpson integration on nested grids together with some integral transformations. In the other more accurate method (see [6], [8] and [9]), the integrand is split into four parts using three optimised break points. Then a fixed-order Gauss-

Legendre quadrature summation is done on the first three parts of the integrand and this is added to the Gauss-Laguerre quadrature done on the last part. Timmes has extended both methods to GFDIs and made their open source FORTRAN routines available from [6] and also used in [7]. Tabulations for a few FDIs may be seen at [1], [3], and [5]. The computational work done for this paper uses instead the precompiled QUADPACK Fortran library which is readily available as a Python SciPy callable function **quad().**

2. Expressions for GFDIs and their Derivatives

All the ten GFDIs, the solutions of which were computed, have been taken from [8]. They are listed again in a compact form below. For convenience, we first define the Fermi-Dirac function $f(x,k,\eta,\beta)$ as the expression under the integral sign in (1.2) as,

$$f(x,k,\eta,\beta) \equiv x^k\sqrt{(1+\ \beta x/2)}/\left(1+exp(x-\eta)\right) \quad (2.1)$$

With this GFDI (1.2) can be rewritten in a compact form as,

$$F_k(\eta,\beta) = \int_0^\infty f(x,k,\eta,\beta)\, dx \quad (2.2)$$

Now using (2.1) we can write all the derivatives compactly starting with the two first order derivative expressions as follows:

$$\partial F_k(\eta,\beta)/\partial\eta = \int_0^\infty \left[f(x,k,\eta,\beta)/\left(1+exp(\eta-x)\right)\right] dx \quad (2.3)$$

$$\partial F_k(\eta,\beta)/\partial\beta = \int_0^\infty f(x,k,\eta,\beta)[\,x/(4+2\beta x)]\, dx \quad (2.4)$$

The three-second order derivative expressions are as follows:

$$\partial^2 F_k(\eta,\beta)/\partial\eta^2 = \int_0^\infty f(x,k,\eta,\beta)\frac{\left(1-\ exp(\eta-x)\right)}{\left(1+exp(\eta-x)\right)^2}\ dx \quad (2.5)$$

$$\partial^2 F_k(\eta,\beta)/\partial\eta\partial\beta = \int_0^\infty f(x,k,\eta,\beta)\left(\frac{x}{4+2\beta x}\right)\left(\frac{1}{1+exp(\eta-x)}\right) dx \quad (2.6)$$

$$\partial^2 F_k(\eta,\beta)/\partial\beta^2 = -\int_0^\infty f(x,k,\eta,\beta)\left(\frac{x}{4+2\beta x}\right)^2\ dx \quad (2.7)$$

Finally, the four third order derivative expressions are as follows:

$$\partial^3 F_k(\eta,\beta)/\partial\eta^3 = \int_0^\infty f(x,k,\eta,\beta)\frac{\left(1-exp(\eta-x)\right)^2-2\,exp(\eta-x)}{\left(1+exp(\eta-x)\right)^3}\ dx \quad (2.8)$$

$$\partial^3 F_k(\eta,\beta)/\partial\eta^2\partial\beta = \int_0^\infty f(x,k,\eta,\beta)\left(\frac{(1-exp(\eta-x))}{(1+exp(\eta-x))^2}\right)\left(\frac{x}{4+2\beta x}\right) dx \quad (2.9)$$

$$\partial^3 F_k(\eta,\beta)/\partial\eta\partial\beta^2 = -\int_0^\infty f(x,k,\eta,\beta)\left(\frac{x}{4+2\beta x}\right)^2\left(\frac{1}{1+exp(\eta-x)}\right) dx \quad (2.10)$$

$$\partial^3 F_k(\eta,\beta)/\partial\beta^3 = \int_0^\infty f(x,k,\eta,\beta)\, 3\, x^3/(4+2\beta x)^3\, dx \quad (2.11)$$

In all the above cases, non-relativistic FDIs are recovered by setting the parameter $\beta = 0$.

3.1 Details of Adaptive Quadrature of GFDIs with quad Module

The professionally developed FORTRAN library QUADPACK some three decades ago [10] has many types of built-in adaptive quadrature algorithms. The SciPy integrator module **quad** is a Python wrapper function providing fast access to all the precompiled double precision FORTRAN routines of QUADPACK package. Hence the quad function seamlessly applies the various needed quadrature methods by analysing the domain of its integration and divides it into subintervals in order to achieve the requested error tolerance in the final result with the added benefit of native compiled speed. Details and more references may be found in [10] and [11]. The quad function can also handle integrals like GFDIs with infinite integration limits. In usage, the quad function returns both the numerical estimate, ***val,*** of the integral and an absolute error estimate, ***err,*** in the integral value:

val, err = *integrate.quad(fdi, a,b, args=(,,),epsrel,absrel,..)* (3.1)

Here, ***fdi*** is one of the ten GFDI functions (Eq. 2.2 to 2.11) to be integrated, **a** and **b** are the lower and upper limits, and ***args*** are the set of constant values passed to the integral. The tolerances for the relative and absolute errors defined by the keyword argument s***epsrel*** and ***epsabs*** have both a default value of $\mathbf{1.49e-08}$ which can be reset with desired values. Now the question is how to achieve the smallest error tolerance possible. The trick [12] that forces the quad function to do so is to set the keyword **epsabs** close **to zero** and set the keyword **epsrel** to the desired value. This is done by setting **epsrel = 2.5e-14** and **epsabs = 0.0** in all the computing work done here. This setting was used to compute the

GFDI expression (2.5) for a set of parameter values given by $\boldsymbol{k} = \mathbf{0.5}$, $\boldsymbol{\beta} = \mathbf{0.0, 0.5, 1.0}$, and $-\mathbf{1} \le \boldsymbol{\eta} \le \mathbf{1}$ for the purpose of comparing the result with the test-run values reported by [10] using a different method.

Table 3.1 Lists of All Values Obtained from the above Code upto 12 Decimal Digits

Values of $\partial^2 F_k(\eta,\beta)/\partial\eta^2 = \int_0^\infty f(x,k,\eta,\beta)\frac{(1-exp(\eta-x))}{(1+exp(\eta-x))^2}\,dx$				
k	$\boldsymbol{\eta}$	$\boldsymbol{\beta} = \mathbf{0.0}$	$\boldsymbol{\beta} = \mathbf{0.5}$	$\boldsymbol{\beta} = \mathbf{1.0}$
0.5	-1.0	2.098711038233e-01	2.526298473929e-01	2.879018677518e-01
0.5	-0.5	2.765653793121e-01	3.388620915154e-01	3.898161363515e-01
0.5	+0.0	3.368591194289e-01	4.234378460735e-01	4.934286856508e-01
0.5	+0.5	3.783041166425e-01	4.925100886273e-01	5.834469366611e-01
0.5	+1.0	3.950134433618e-01	5.380343605687e-01	6.498435028375e-01
Abs Errors:		4.3855302e-16	7.7715612e-16	7.2147122e-16

These values are in complete agreement with all the test-run values reported but only up to six decimal digits [10]. The quad module generally estimates the absolute error in the integral by computing it using two related methods such as the 30-point Gauss rule and the 61-point Kronrod rule and finding the difference between them. If the accuracy is low, the interval is bisected and the process repeats for sub-intervals. The last row in Table 3.1 shows that the absolute error in the computations as internally estimated by the quad function is slightly lesser than an order of magnitude higher than the machine precision.

In other computational comparison trials, it was found possible to reproduce all the tabular values listed by Cloutman (see [5]) and also those listed by Clayton (see [3]) with increased accuracy and a greater number of decimal digits.

The main project of this paper is to integrate all the ten GFDIs defined in Sec. 2 using the adaptive quadrature integrator ***quad***, draw separate plots of all of them and also provide a short table of values for comparison purposes. It was also decided to compute the three optimised break points as prescribed by Aparicio in Appendix B of [8] and use them as limits in quad so that it now integrates each GFDI in four separate parts which are then added together. A callable Python function script written to compute the three break points is listed in Table A.1 of Appendix A along with all the required constants. The same breakpoint computation has been adopted by [6] and [8] to carry out a four-part integration of GFDIs using a few fixed-order quadratures for guaranteed accuracy. However, in this study, the aforementioned **quad** function is employed so that its performance can be evaluated and compared with results obtained by [6] in a level playing field. The range of parameters for each GFDI has been specified as follows:

Each computation is done once with the relativity parameter $\boldsymbol{\beta = 0}$ and then repeated with $\boldsymbol{\beta = 1}$ **for each** of the nine values of the order parameter $\boldsymbol{k}$ ranging from $\boldsymbol{-0.5\ to\ 5.0}$ while the degeneracy parameter $\boldsymbol{\eta}$ is varied from ***-20.0*** to ***+20.0.***

An example Python script given in Table A.2 was written for integrating the GFDI given by Eq.2.2 in four parts by computing the $\boldsymbol{\eta}$ dependent break points and for the printing of the three function values only for $\boldsymbol{\eta = -20.0,\ 0.0,\ 20.0}$ each $\boldsymbol{k}$ parameter. Table A.2 also includes a script for plotting the solutions for each GFDI evaluated for both values of the relativity parameter. Thus, each figure frame contains 9 pairs of curves with each pair of the same colour in a solid line and dashed line format. The results have been presented in ten figures with each accompanied by a short table of just three function values. The semi-log graphs are presented sequentially as Fig. 3.1 to Fig. 3.10 and the data are presented in Table 3.2 to Table 3.11.

3.2 Discussion of the Results

The graphs in the first six figures Fig. 3.1 to Fig. 3.6 representing GFDIs for both values of the relativity parameter, their two first order derivatives and three-second order derivatives (all with respect to $\boldsymbol{\eta}$ and $\boldsymbol{\beta}$) bear excellent comparison with the six figures shown in [6]. The remaining four figures Fig. 3.7 to Fig. 3.10 respectively represent the four third order partial derivatives.

Now consider Table 3.2 listing all values to double precision (DP) of 16 decimal digits. In this table, the nine value pairs obtained for the GFD function are given first and are followed by the starred row which lists the DP values for $\boldsymbol{\beta = 1}$ (as deducted by Timmes

[6]) for comparing with the row preceding it. The rows to be compared are in a bigger font and the differing digits in the corresponding values are underlined. In the third column, all the differing digits have a place value less than 10^{-22} and hardly affect DP level calculations. These low values in the third column increase over a billion folds to those seen in the fourth column. These values, in turn, rise more moderately toward the values in the fifth column along with the increasing order parameter. In the fourth and fifth columns, many value pairs are seen to be identical and the difference is very small. These may be due to differences in the behaviour of the derivative functions themselves to which the adaptive quadrature has internally responded as well as due to LSB errors. The same comments apply also to all the remaining nine tables following it and they list values up to 14 decimal digits only in order to omit uncertainties in the last two digits of full DP values.

Table 3.2: (Refer Fig. 3.1) Fermi-Dirac Function (GFDI) (erel=2.5e-14; eabs=0.0)

Eq. 2.2: $F_k(\eta, \beta) = \int_0^\infty x^k \sqrt{(1+\beta x/2)}/(1+exp(x-\eta))\, dx$
Method: Aparicio+quad()

k	β	$\eta = -20$	$\eta = 0$	$\eta = 20$	
-0.5	0	3.6532996700745490e-09	1.0721549299401900	8.9349726661669671	
-0.5	1	4.052625467814804 1e-9	1.218129524589 5984	17.472106444954584	
-.5 ★	1	4.052625467814804 9E-9	1.218129524589 6004	17.472106444954584	★
0.5	0	1.8266498363684070e-09	0.67809389515310092	59.812795370358018	
0.5	1	2.38462122879289 11e-9	0.910058252873233 50	155.58338998028 805	
0.5 ★	1	2.38462122879289 23E-9	0.910058252873233 61	155.58338998028 799	★
1.0	0	2.0611536213764700e-09	0.82246703342411331	201.64493406478709	
1.0	1	2.8746037540323 896e-9	1.174763658234733 2	553.80766697187244	
1 ★	1	2.8746037540323 904E-9	1.174763658234733 4	553.80766697187244	★
1.5	0	2.7399747555509587e-09	1.1528038370883613	726.56828396517528	
1.5	1	4.05262547218346 17e-9	1.7388257662295030	2068.6869148416063	
1.5 ★	1	4.05262547218346 34E-9	1.7388257662295030	2068.6869148416063	★
2.0	0	4.1223072438150282e-09	1.8030853547393912	2732.4640293447178	
2.0	1	6.428115834475764 0e-9	2.8558702644518101	7983.4608510845 892	
2 ★	1	6.428115834475764 9E-9	2.8558702644518101	7983.4608510845 901	★
2.5	0	6.8499368901253336e-09	3.0825860828374183	10590.639176614390	
2.5	1	1.120648916278667 1e-8	5.1048722017474431	31554.778676830 399	
2.5★	1	1.120648916278667 3E-8	5.1048722017474431	31554.778676830 407	★

3.0	0	1.2366921733038218e-08	5.6821969769834757	41985.285274159483	
3.0	1	2.11405887728698 87e-8	9.8046085338899 189	127047.9188473585 4	
3 ★	1	2.11405887728698 97E-8	9.8046085338899 207	127047.9188473585 8	★
4.0	0	4.9467686935339126e-08	23.330874490725826	693547.04165551020	
4.0	1	9.14201669482467 02e-8	43.3472545608483 91	2148146.41125072 40	
4 ★	1	9.14201669482467 41E-8	43.3472545608483 84	2148146.41125072 54	★
5.0	0	2.4733843468466131e-07	118.26613095569222	12028308.028222781	
5.0	1	4.89095294523815 74e-7	234.5490884049686 7	37898779.2505643 22	
5 ★	1	4.89095294523815 96E-7	234.5490884049686 1	37898779.2505643 52	★
Rows with ★are DP values from Timmes [6] and listed here for comparison★ ★Methods of [6] and [8] : Aparicio + Fixed-orderGaussian quadratures★					

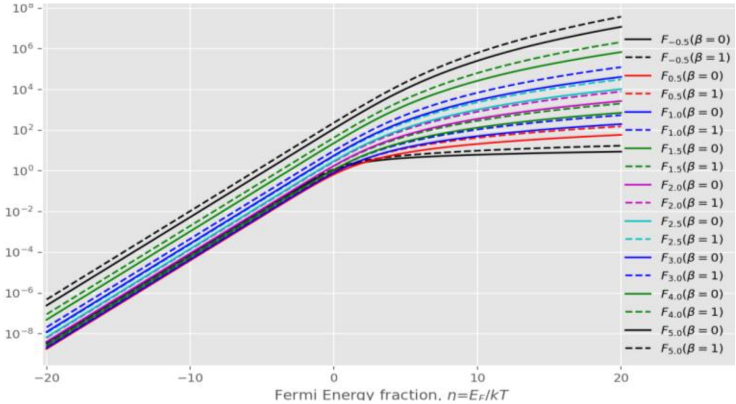


Figure 3.1: Note: Curve Labels Apply to Curve Pairs from Bottom Up (see Table 3.2)

Table 3.3: (Refer Fig. 3.2) 1st Order Partial Derivative of GFDI

Eq. 2.3: $\partial F_k(\eta,\beta)/\partial\eta = \int_0^\infty [f(x,k,\eta,\beta)/(1+exp(\eta-x))]\,dx$				
k	β	$\eta=-20$	$\eta=0$	$\eta=20$
-0.5	0	3.65329966475003e-09	6.73718238857754e-01	2.24315128619932e-01
-0.5	1	4.05262546218209e-09	7.87409006276875e-01	7.41900547225267e-01
0.5	0	1.82664983503728e-09	5.36077464970096e-01	4.46748633308349e+00
0.5	1	2.38462122724059e-09	7.38544721734421e-01	1.48322666496613e+01
1.0	0	2.06115362031438e-09	6.93147180559945e-01	2.00000000020612e+01
1.0	1	2.87460375273975e-09	1.01171822023570e+00	6.65245097366169e+01
1.5	0	2.73997475455261e-09	1.01714084272965e+00	8.97191930555371e+01
1.5	1	4.05262547091941e-09	1.56115658686723e+00	2.98974353106208e+02
2.0	0	4.12230724275294e-09	1.64493406684823e+00	4.03289868129574e+02

2.0	1	6.42811583308068e-09	2.64152241074626e+00	1.34633957623024e+03
2.5	0	6.84993688887740e-09	2.88200959272090e+00	1.81642070991294e+03
2.5	1	1.12064891610902e-08	4.82433888357376e+00	6.07486220825327e+03
3.0	0	1.23669217314451e-08	5.40925606421817e+00	8.19739208803416e+03
3.0	1	2.11405887706334e-08	9.41155573702776e+00	2.74646644942231e+04
4.0	0	4.94676869321529e-08	2.27287879079339e+01	1.67941141096638e+05
4.0	1	9.14201669435023e-08	4.24328485151600e+01	5.64687070931175e+05
5.0	0	2.47338434676696e-07	1.16654372453629e+02	3.46773520827755e+06
5.0	1	4.89095294511311e-07	2.31981071325466e+02	1.17018144707955e+07

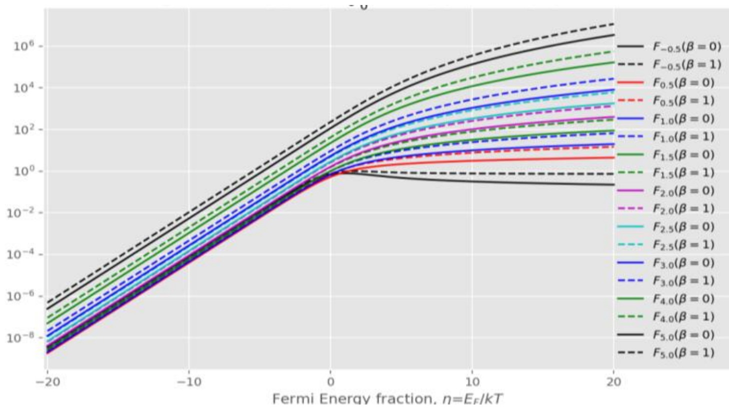


Figure 3.2: Note: Curve Labels Apply to Curve Pairs from Bottom Up (See Table 3.3).

Table 3.4: (refer Fig. 3.3) 1st Order Partial Derivative of GFDI

Eq. 2.4: $\partial F_k(\eta,\beta)/\partial\beta = \int_0^\infty f(x,k,\eta,\beta)[\,x/(4+2\beta x)]\,dx$				
k	β	$\eta = -20$	$\eta = 0$	$\eta = 20$
-0.5	0	4.56662459092102e-10	1.69523473788275e-01	1.49531988425895e+01
-0.5	1	3.58308493333187e-10	1.29479959439621e-01	6.09621342718404e+00
0.5	0	6.84993688887740e-10	2.88200959272090e-01	1.81642070991294e+02
0.5	1	4.75693627730073e-10	1.96069207557375e-01	6.55992681357759e+01
1.0	0	1.03057681095376e-09	4.50771338684848e-01	6.83116007336179e+02
1.0	1	6.78908325015898e-10	2.91995094276789e-01	2.38724242286490e+02
1.5	0	1.71248422253133e-09	7.70646520709355e-01	2.64765979415360e+03
1.5	1	1.07492548063159e-09	4.77274468000001e-01	9.03144921149251e+02
2.0	0	3.09173043325955e-09	1.42054924424587e+00	1.04963213185399e+04
2.0	1	1.85624126720609e-09	8.43944943672327e-01	3.51428194096931e+03
2.5	0	5.99369477940564e-09	2.79592918792333e+00	4.23548243087561e+04
2.5	1	3.45339362013016e-09	1.59788716487372e+00	1.39710994961167e+04
3.0	0	1.23669217338348e-08	5.83271862268145e+00	1.73386760413878e+05

3.0	1	6.85781185202277e-09	3.21441437960030e+00	5.64953955417406e+04
4.0	0	6.18346086711653e-08	2.95665327389231e+01	3.00707700705569e+06
4.0	1	3.19944597700778e-08	1.52447985212236e+01	9.61082414541881e+05
5.0	0	3.71007652032966e-07	1.78666887586109e+02	5.40716409611736e+07
5.0	1	1.80558727721752e-07	8.67849471600371e+01	1.70272247961984e+07

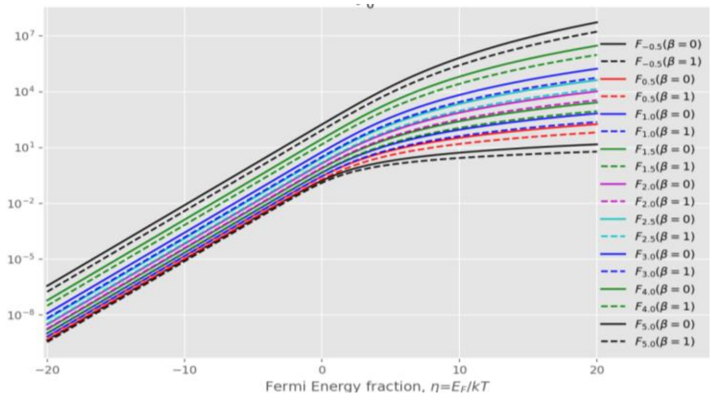


Figure 3.3: Note: Curve Labels Apply to Curve Pairs from Bottom Up (See Table 3.4).

Table 3.5: (Refer Fig. 3.4) 2nd Order Partial Derivative of GFDI

Eq. 2.5: $\partial^2 F_k(\eta,\beta)/\partial\eta^2 = \int_0^\infty f(x,k,\eta,\beta)\frac{(1-exp(\eta-x))}{(1+exp(\eta-x))^2}\,dx$				
k	β	$\eta = -20$	$\eta = 0$	$\eta = 20$
-0.5	0	3.65329965410099e-09	2.10356366336200e-01	-5.68073326041743e-03
-0.5	1	4.05262545091665e-09	2.79208238045097e-01	-1.72771993985022e-03
0.5	0	1.82664983237502e-09	3.36859119428877e-01	1.12157564309966e-01
0.5	1	2.38462122413598e-09	4.93428685650831e-01	7.07928965343908e-01
1.0	0	2.06115361819020e-09	5.00000000000000e-01	9.99999997938851e-01
1.0	1	2.87460375015446e-09	7.63876584311700e-01	4.81952267751162e+00
1.5	0	2.73997475255591e-09	8.04116197455143e-01	6.70122949962524e+00
1.5	1	4.05262546839130e-09	1.27764107844405e+00	2.89905759158602e+01
2.0	0	4.12230724062876e-09	1.38629436111989e+00	4.00000000041224e+01
2.0	1	6.42811583029050e-09	2.28586144095327e+00	1.63299790451121e+02
2.5	0	6.84993688638153e-09	2.54285210682413e+00	2.24297982638843e+02
2.5	1	1.12064891576974e-08	4.34382176891684e+00	8.83438707435889e+02
3.0	0	1.23669217282588e-08	4.93480220054468e+00	1.20986960438872e+03
3.0	1	2.11405887661603e-08	8.72047843664816e+00	4.65168697485006e+03
4.0	0	4.94676869257803e-08	2.16370242568727e+01	3.27895683521367e+04

4.0	1	9.14201669340135e-08	4.07601784078061e+01	1.22365653731685e+05
5.0	0	2.47338434660764e-07	1.13643939539670e+02	8.39705705483190e+05
5.0	1	4.89095294486303e-07	2.27152755913990e+02	3.08076489861188e+06

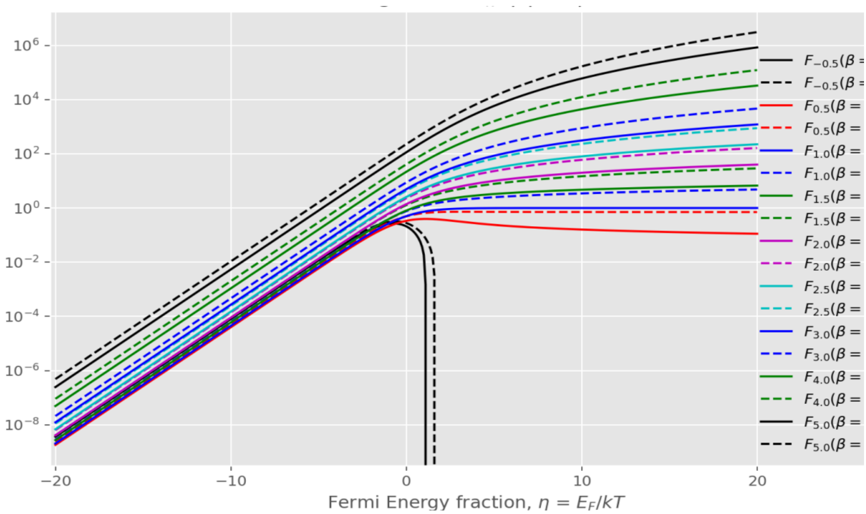


Figure 3.4: Note: Curve Labels Apply to Curve Pairs From Bottom Up (See Table 3.5).

Table 3.6: (Refer Fig. 3.5) 2nd Order Partial Derivative of GFDI

Eq. 2.6: $\partial^2 F_k(\eta,\beta)/\partial\eta\partial\beta = \int_0^{\infty} f(x,k,\eta,\beta)\left(\frac{x}{(4+2\beta x)}\right)\left(\frac{1}{1+exp(\eta-x)}\right)dx$				
k	β	$\eta = -20$	$\eta = 0$	$\eta = 20$
-0.5	0	4.56662458759319e-10	1.34019366242524e-01	1.11687158327087e+00
-0.5	1	3.58308493044936e-10	9.97241825123948e-02	3.36978691731272e-01
0.5	0	6.84993688638153e-10	2.54285210682413e-01	2.24297982638843e+01
0.5	1	4.75693627530422e-10	1.69823995842421e-01	6.74217594136812e+00
1.0	0	1.03057681068823e-09	4.11233516712057e-01	1.00822467032394e+02
1.0	1	6.78908324811013e-10	2.62425000481866e-01	3.02507709778872e+01
1.5	0	1.71248422221935e-09	7.20502398180226e-01	4.54105177478235e+02
1.5	1	1.07492548039886e-09	4.40930301748771e-01	1.36002824670368e+02
2.0	0	3.09173043286127e-09	1.35231401605454e+00	2.04934802200854e+03
2.0	1	1.85624126691831e-09	7.95911204409397e-01	6.12668246159343e+02
2.5	0	5.99369477885966e-09	2.69726282248274e+00	9.26680927953759e+03
2.5	1	3.45339361974740e-09	1.53030883828934e+00	2.76542545478590e+03
3.0	0	1.23669217330382e-08	5.68219697698347e+00	4.19852852741595e+04
3.0	1	6.85781185148006e-09	3.11395545969509e+00	1.25069957547929e+04
4.0	0	6.18346086691739e-08	2.91635931134073e+01	8.66933802069388e+05

4.0	1	3.19944597687910e-08	1.49885133381898e+01	2.57329543956002e+05
5.0	0	3.71007652026992e-07	1.77399196433538e+02	1.80424620423342e+07
5.0	1	1.80558727718074e-07	8.60135089863531e+01	5.33624814748575e+06

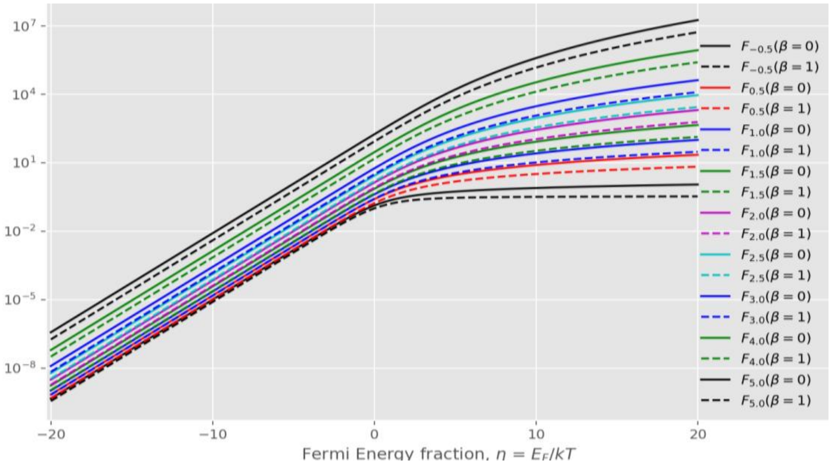


Figure 3.5: Note: Curve Labels Apply to Curve Pairs from Bottom Up (See Table 3.6)

Table 3.7: (Refer Fig. 3.6) 2nd Order Partial Derivative of GFDI

Eq. 2.7: $\partial^2 F_k(\eta,\beta)/\partial\beta^2 = -\int_0^\infty f(x,k,\eta,\beta)[\,x/(4+2\beta x)]^2\,dx$				
k	β	$\eta = -20$	$\eta = 0$	$\eta = 20$
-0.5	0	1.71248422221935e-10	7.20502398180226e-02	4.54105177478234e+01
-0.5	1	6.17691124693574e-11	2.43959433170100e-02	2.40214419940794e+00
0.5	0	4.28121055632833e-10	1.92661630177339e-01	6.61914948538399e+02
0.5	1	1.14308588926322e-10	4.92427171446676e-02	2.79953456690721e+01
1.0	0	7.72932608314888e-10	3.55137311061467e-01	2.62408032963497e+03
1.0	1	1.80483708129381e-10	8.00740784209705e-02	1.03504480700128e+02
1.5	0	1.49842369485141e-09	6.98982296980832e-01	1.05887060771890e+04
1.5	1	3.08845562463150e-10	1.40151799710665e-01	3.95581769236481e+02
2.0	0	3.09173043345869e-09	1.45817965567036e+00	4.33466901034694e+04
2.0	1	5.67153217344280e-10	2.61824314994223e-01	1.55013200908440e+03
2.5	0	6.74290662713845e-09	3.20565658145022e+00	1.79592678839064e+05
2.5	1	1.10900568513878e-09	5.18639983015529e-01	6.19438620958538e+03
3.0	0	1.54586521677913e-08	7.39163318473076e+00	7.51769251763924e+05
3.0	1	2.29459949132283e-09	1.08355855981171e+00	2.51474337527015e+04
4.0	0	9.27519130082415e-08	4.46667218965274e+01	1.35179102402934e+07
4.0	1	1.14080309023933e-08	5.45528214098837e+00	4.30246339765537e+05
5.0	0	6.49263391062918e-07	3.13813395583584e+02	2.50160498819933e+08
5.0	1	6.74633020560896e-08	3.24819092980418e+01	7.65311971856812e+06

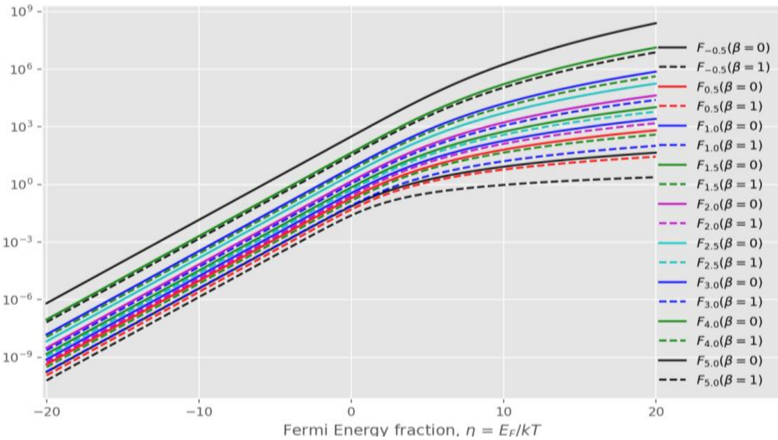


Figure 3.6: Note: Curve Labels Apply to Curve Pairs from Bottom Up (See Table 3.7).

Table 3.8: (Refer Fig. 3.7) 3rd Order Partial Derivative of GFDI

Eq. 2.8 : $\partial^3 F_k(\eta,\beta)/\partial\eta^3 = \int_0^\infty f(x,k,\eta,\beta) \frac{(1-exp(\eta-x))^2 - 2\,exp(\eta-x)}{(1+exp(\eta-x))^3}\,dx$				
k	β	$\eta = -20$	$\eta = 0$	$\eta = 20$
-0.5	0	3.65329963280290e-09	-1.55694332690436e-01	4.35603884270473e-04
-0.5	1	4.05262542838583e-09	-1.38004155239571e-01	1.73260547393172e-04
0.5	0	1.82664982705049e-09	1.05178183168100e-01	-2.84036663020828e-03
0.5	1	2.38462121792677e-09	1.98326384377750e-01	-8.03802871999493e-05
1.0	0	2.06115361394185e-09	2.50000000000000e-01	2.06115492729086e-09
1.0	1	2.87460374498389e-09	4.33265505517399e-01	1.16833034054836e-01
1.5	0	2.73997474856252e-09	5.05288679143316e-01	1.68236346464949e-01
1.5	1	4.05262546333510e-09	8.69412840591260e-01	1.41429097132236e+00
2.0	0	4.12230723638040e-09	1.00000000000000e+00	1.99999999587769e+00
2.0	1	6.42811582471016e-09	1.74252315967332e+00	1.19662890507315e+01
2.5	0	6.84993688138979e-09	2.01029049363786e+00	1.67530737490631e+01
2.5	1	1.12064891509117e-08	3.57438361110213e+00	8.62669327009673e+01
3.0	0	1.23669217218863e-08	4.15888308335967e+00	1.20000000012367e+02
3.0	1	2.11405887572143e-08	7.57097506123660e+00	5.66894779187506e+02
4.0	0	4.94676869130353e-08	1.97392088021787e+01	4.83947841755488e+03
4.0	1	9.14201669150360e-08	3.78153714881631e+01	2.07786005711569e+04
5.0	0	2.47338434628902e-07	1.08185121284363e+02	1.63947841760683e+05
5.0	1	4.89095294436285e-07	2.18314065759793e+02	6.68667390180754e+05

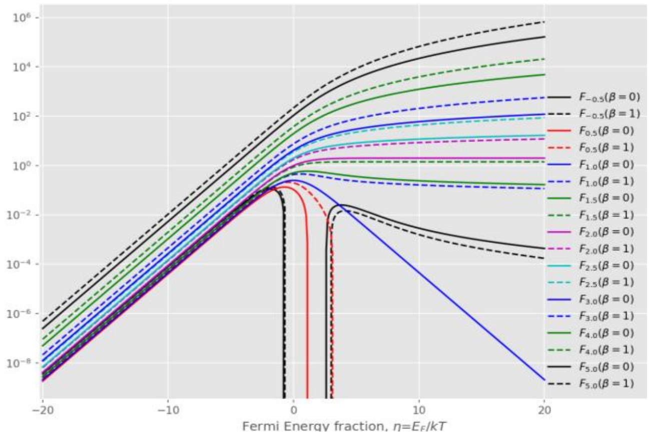


Figure 3.7: Note: Curve Labels Apply to Curve Pairs from Bottom Up. Also, notice the behaviour of curve pairs for k=-0.5 (black) and for k=0.5 (red) at the bottom. See Table 3.8 for some values.

Table 3.9: (Refer Fig. 3.8) 3rd Order Partial Derivative of GFDI

Eq. 2.9: $\partial^3 F_k(\eta,\beta)/\partial\eta^2\partial\beta = \int_0^\infty f(x,k,\eta,\beta)\left(\frac{(1-exp(\eta-x))}{(1+exp(\eta-x))^2}\right)\left(\frac{x}{(4+2\beta x)}\right) dx$

k	β	$\eta=-20$	$\eta=0$	$\eta=20$
-0.5	0	4.56662458093754e-10	8.42147798572192e-02	2.80393910774916e-02
-0.5	1	3.58308492468435e-10	5.87222653552015e-02	7.83479682724530e-04
0.5	0	6.84993688138979e-10	2.01029049363786e-01	1.67530737490631e+00
0.5	1	4.75693627131122e-10	1.29269812115013e-01	3.52397523306504e-01
1.0	0	1.03057681015719e-09	3.46573590279973e-01	1.00000000010306e+01
1.0	1	6.78908324401242e-10	2.14769991049915e-01	2.32724369570828e+00
1.5	0	1.71248422159538e-09	6.35713026706032e-01	5.60744956597108e+01
1.5	1	1.07492547993341e-09	3.80280914992001e-01	1.37904929113171e+01
2.0	0	3.09173043206470e-09	1.23370055013617e+00	3.02467401097181e+02
2.0	1	1.85624126634277e-09	7.13390738376802e-01	7.69954078341441e+01
2.5	0	5.99369477776772e-09	2.52175839363079e+00	1.58936812117382e+03
2.5	1	3.45339361898187e-09	1.41134905447442e+00	4.14138367895311e+02
3.0	0	1.23669217314451e-08	5.40925606421817e+00	8.19739208803416e+03
3.0	1	6.85781185039463e-09	2.93345774157048e+00	2.17185267175674e+03
4.0	0	6.18346086651911e-08	2.84109848849174e+01	2.09926426370798e+05
4.0	1	3.19944597662175e-08	1.45131737207621e+01	5.68391215223292e+04
5.0	0	3.71007652015043e-07	1.74981558680444e+02	5.20160281241633e+06
5.0	1	1.80558727710716e-07	8.45500305154707e+01	1.42670420626128e+06

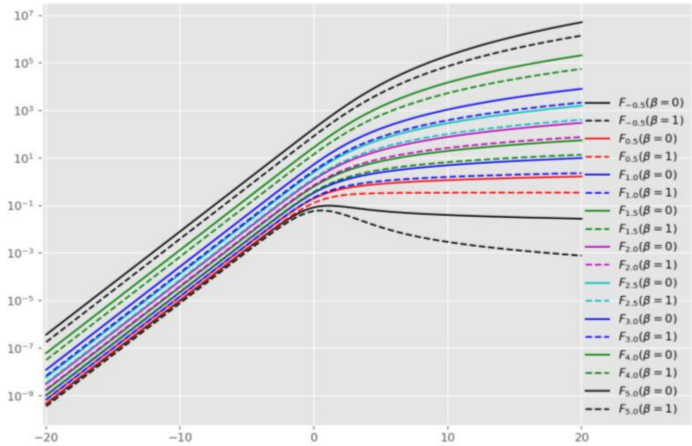


Figure 3.8: Note: Curve Labels Apply to Curve Pairs from Bottom Up (See Table 3.9).

Table 3.10: (Refer Fig. 3.9) 3rd Order Partial Derivative of GFDI

Eq. 2.10: $\partial^3 F_k(\eta,\beta)/\partial\eta\partial\beta^2 = -\int_0^\infty f(x,k,\eta,\beta)\left(\frac{x}{(4+2\beta x)}\right)^2\left(\frac{1}{1+exp(\eta-x)}\right) dx$				
k	β	$\eta = -20$	$\eta = 0$	$\eta = 20$
-0.5	0	1.71248422159538e-10	6.35713026706032e-02	5.60744956597107e+00
-0.5	1	6.17691124362819e-11	2.03164616535794e-02	1.53070514290404e-01
0.5	0	4.28121055554837e-10	1.80125599545056e-01	1.13526294369559e+02
0.5	1	1.14308588892647e-10	4.42790746140517e-02	3.06494694210325e+00
1.0	0	7.72932608215317e-10	3.38078504013636e-01	5.12337005502135e+02
1.0	1	1.80483708090269e-10	7.38842630687946e-02	1.37569050995175e+01
1.5	0	1.49842369471492e-09	6.74315705620685e-01	2.31670231988440e+03
1.5	1	3.08845562414135e-10	1.31907001646282e-01	6.18715184509774e+01
2.0	0	3.09173043325955e-09	1.42054924424587e+00	1.04963213185399e+04
2.0	1	5.67153217278619e-10	2.50187076067109e-01	2.78820312880637e+02
2.5	0	6.74290662683134e-09	3.14542033641375e+00	4.76491773473507e+04
2.5	1	1.10900568504543e-09	5.01340415852105e-01	1.25896969049099e+03
3.0	0	1.54586521672935e-08	7.29089827835182e+00	2.16733450517347e+05
3.0	1	2.29459949118279e-09	1.05660357771333e+00	5.69585725163516e+03
4.0	0	9.27519130067480e-08	4.43497991083846e+01	4.51061551058355e+06
4.0	1	1.14080309020299e-08	5.38104951366826e+00	1.17273057474731e+05
5.0	0	6.49263391057691e-07	3.12667053275692e+02	9.46253716820538e+07
5.0	1	6.74633020549769e-08	3.22446554658400e+01	2.43357795879341e+06

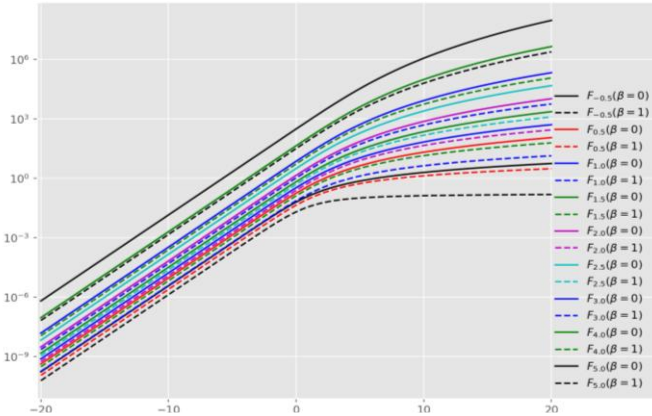
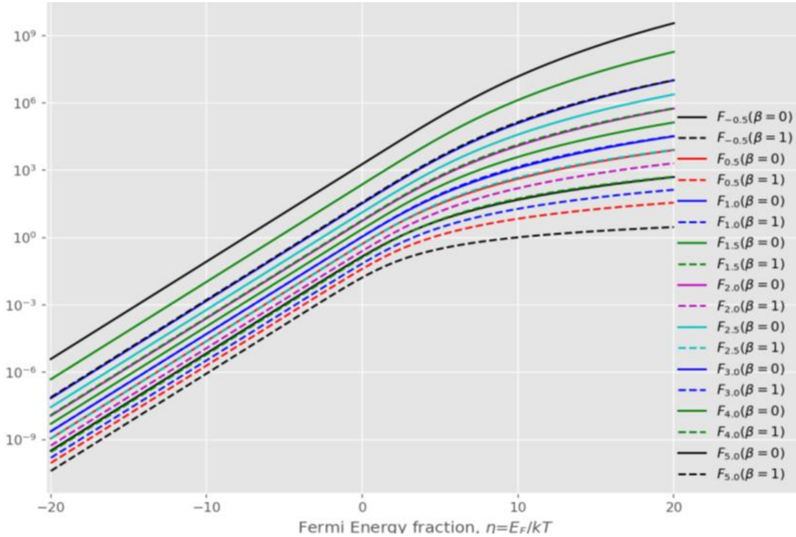


Figure 3.9: Note: curve labels apply to curve pairs from bottom up. (See Table 3.10).

Table 3.11: (Refer Fig. 3.10) 3rd Order Partial Derivative of GFDI

Eq. 2.11: $\partial^3 F_k(\eta,\beta)/\partial\beta^3 = \int_0^\infty f(x,k,\eta,\beta)\,3\,x^3/(4+2\beta x)^3\,dx$				
K	β	$\eta=-20$	$\eta=0$	$\eta=20$
-0.5	0	3.21090791724625e-10	1.44496222633004e-01	4.96436211403799e+02
-0.5	1	4.01141922807460e-11	1.67107836784734e-02	2.94041355641660e+00
0.5	0	1.12381777113856e-09	5.24236722735624e-01	7.94152955789178e+03
0.5	1	9.12344988279904e-11	4.04425083600546e-02	3.61121913907749e+01
1.0	0	2.31879782509402e-09	1.09363474175277e+00	3.25100175776020e+04
1.0	1	1.54781615238906e-10	7.01092376167730e-02	1.35197609919877e+02
1.5	0	5.05717997035384e-09	2.40424243608766e+00	1.34694509129298e+05
1.5	1	2.80799346038744e-10	1.29342682845889e-01	5.21148271073172e+02
2.0	0	1.15939891258435e-08	5.54372488854807e+00	5.63826938822943e+05
2.0	1	5.41166595538608e-10	2.52517997257788e-01	2.05480279378685e+03
2.5	0	2.78144898375795e-08	1.33548655777278e+01	2.38140507818225e+06
2.5	1	1.10190983563069e-09	5.19274608831517e-01	8.24928277223173e+03
3.0	0	6.95639347561811e-08	3.35000414223955e+01	1.01384326802201e+07
3.0	1	2.35956604590702e-09	1.12030184520199e+00	3.36115450414786e+04
4.0	0	4.86947543297189e-07	2.35360046687688e+02	1.87620374114950e+08
4.0	1	1.23929142617758e-08	5.94231952107859e+00	5.78146419565349e+05
5.0	0	3.89558034639319e-06	1.88639822235363e+03	3.55673842068222e+09
5.0	1	7.64091245605828e-08	3.68382249049056e+01	1.03233867387215e+07



4. Conclusion

The use of the automatically adaptive integrator quad combined with optimised break points as demonstrated above is convenient for fast and accurate computation of all Fermi-Dirac integrals. It is thus a useful alternative for the use of fixed-order quadrature schemes in many stellar EOS calculations and in other applications.

Acknowledgement

The author would like to thank F.X. Timmes of Arizona State University for the information shared on his webpage [6] on Fermi-Dirac integrals and to him personally for sharing requested values accurate to DP and QP levels which enabled a comparison between the quadrature methods employed by him and this paper.

References

[1] J. McDougall and E.C. Stoner, Phil. Trans. Roy. Soc. London, A, Vol 231, 67, 1939.

[2] R.B. Larson, and P.R. Demarque, An application of Henyey's approach to the integration of the equations of stellar structure, The Astrophysical Journal, 140, 524, 1964.

[3] D.D. Clayton, Principles of stellar evolution and nucleosynthesis, McGraw-Hill, 1968
[4] J.P. Cox and R.T. Giuli, 1968, Principles of stellar structure, Gordon & Breach, 1968.
[5] L.D. Cloutman, Numerical evaluation of FDIs, Ap. J. Supp., 7, 677-699, 1989.
[6] F.X.Timmes, http:// cococubed.asu.edu/ code_pages/ fermi_dirac.shtml
[7] F.X.Timmes, http://cococubed.asu.edu/code_pages/eos.shtml[
[8] Z. Gong etal., "Generalized FD Functions and Derivatives: Properties and Evaluation", arXiv:astro-ph/0102329v1.20Feb.2001.
[9] http://www.cpc.cs.qub.ac.uk/summaries/ADNX_v1_0.html
[10] http://nines.cs.kuleuven.be/software/QUADPACK/
[11] https:// github.com /scipy/scipy/ tree/master/scipy/ integrate/ quadpack
[12] J.D. Cook, https://www.johndcook.com/blog/2012/ 03/20/scipy-integration/

Appendix A

Table A.1: Python script to compute Aparicio's break points

This Filename : S1S2S3Calc.py
# -*- coding: utf-8 -*-
import scipy as np
# Reference: Appendix B of Paper by ZGong et al., 2001
#Download pdf from arXiv:astro-ph/01022329v1 20 Feb 2001
this script was written by: M.N. Anandaram
def break_points (eta) :
# Data for computing break points S1, S2 and S3 :
D = np.array([3.36090, 4.995510, 3.938300, 4.174440])
sgma = np.array([0.091186, 0.0911856, 0.0911856, 0.0911856])
a1 = np.array([6.77740 , 6.777400, 6.777400 , 6.777400])
b1 = np.array([1.14180 , 1.141800, 1.141800 , 1.141800])
c1 = np.array([2.98260 , 2.982550, 2.982550 , 2.982550])
a2 = np.array([3.76010 , 3.760100, 3.760100 , 3.760100])
b2 = np.array([0.093719, 0.0937188, 0.0937188, 0.0937188])
c2 = np.array([0.021064, 0.0210635, 0.0210635, 0.0210635])
d2 = np.array([31.0840 , 39.50150 , 31.44990 , 30.54120])
e2 = np.array([1.00560 , 1.005570, 1.005570 , 1.005570])
a3 = np.array([7.56690 , 7.566900, 7.566900 , 7.566900])
b3 = np.array([1.16950 , 1.169530, 1.169530 , 1.169530])
c3 = np.array([0.75416 , 7.541620, 7.541620 , 7.541620])
d3 = np.array([6.65590 , 7.647340, 6.863460 , 7.880300])
e3 = np.array([-1.28190 , -0.128190, -0.128190 , -0.128190])

xieta = np.log(1.0 + np.exp(sgma*(eta - D)))/sgma
xi2 = xieta * xieta # Note: - xieta is a 4 element array
xa = (a1 + b1*xieta + c1*xi2)/(1.0 + c1*xieta)
xb = (a2 + b2*xieta + c2*d2*xi2)/(1.0 + e2*xieta + c2*xi2)
xc = (a3 + b3*xieta + c3*d3*xi2)/(1.0 + e3*xieta + c3*xi2)
S1, S2, S3 = xa - xb, xa, xa + xc
return S1,S2,S3 # Note: S1,S2,S3 are four element arrays
#if __name__ == '__main__': # End of Table A.1

Table A.2: Python Script for quad () Integration with break points

0	Filename : quad_FDk_eta_bta.py
1	# -*- coding: utf-8 -*- import scipy as np from scipy.integrate import quad from matplotlib import pyplot as plt

2	# define the generalized Fermi-Dirac function
	def FDk_Eta_Bta(x, k, eta, beta) : # FD(x, k, Eta, Bta)
	xFermi = x**k / (np.exp(x – eta) + 1.0)
	return xFermi*(1.0 + 0.5*beta*x)**0.5 # FD (x,k,eta, beta > 0)

3	# Function to calculate Aparicio's break points S1, S2 and S3 # All three have four parameters each needed for all 10 GFDIs. # For example, s10, s20 and s30 are break points for Eq. 2.2 only. # s11, s21 and s31 are brk pnts for Eq. 2.3 and Eq. 2.4 # s12, s22 and s32 are brk pnts for Eq. 2.5, Eq. 2.6 and Eq. 2.7 # s13, s23 and s33 are brk pnts for Eqns. 2.8, 2.9, 2.10 and 2.11
4	def getS123(xeta) : from S1S2S3Calc import break_points s1, s2, s3 = break_points (xeta) s10,s11,s12,s13 = s1[0], s1[1], s1[2], s1[3] # unravel all 4 parts s20,s21,s22,s23 = s2[0], s2[1], s2[2], s2[3] s30,s31,s32,s33 = s3[0], s3[1], s3[2], s3[3] return s10, s20, s30 # break points for FDI function only
5	#
6	# set the parameters
7	a = 0.0 # lower limit of integration
8	b = np.inf # upper limit of integration
9	N = 401 # number of Eta values to use in integrations
10	#k is order parameter (exponent k of x as in x**k)
11	eta_array = np.linspace (-20.0, 20.0, N) # array of values of Eta
12	kvals = [-0.5, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 4.0, 5.0];
13	clors = ['k', 'r', 'b', 'g', 'm', 'c', 'b', 'g', 'k']
14	plt.figure(1, figsize = (10, 7), dpi = 180)
15	erl = 2.5e-14 # settable value of keyword epsrel
16	for k, clr in zip (kvals, clors) :
17	val1 = np.zeros(N, float) # array to store N results

```
    val2 =  np.zeros(N,  float)  # array to store N  results
    # do the 4-part integrations with  integrate.quad()
    idx = 0 # start of array index for storage allocation
    for eta in eta_array:
        s1d0, s2d0, s3d0 = getS123(eta)
        #print(s1d0, s2d0, s3d0)
        val1A, err = quad(FD_Eta_Bta,   a  , s1d0,
                                  args=(k,eta,0.0),   epsrel=erl, epsabs=0.0)
        val1B, err = quad(FD_Eta_Bta,  s1d0,  s2d0,
                                  args=(k,eta,0.0),   epsrel=erl, epsabs=0.0)
        val1C, err = quad(FD_Eta_Bta,  s2d0,  s3d0,
                                  args=(k,eta,0.0),   epsrel=erl, epsabs=0.0)
        val1D, err = quad(FD_Eta_Bta, s3d0,  b ,
                                  args=(k,eta,0.0),   epsrel=erl, epsabs=0.0)
        val2A, err = quad(FD_Eta_Bta,   a , s1d0,
                                  args=(k,eta,1.0),   epsrel=erl, epsabs=0.0)
        val2B, err = quad(FD_Eta_Bta,  s1d0,  s2d0,
                                  args=(k,eta,1.0),   epsrel=erl, epsabs=0.0)
        val2C, err = quad(FD_Eta_Bta,  s2d0, s3d0,
                                  args=(k,eta,1.0),   epsrel=erl, epsabs=0.0)
        val2D, err = quad(FD_Eta_Bta,  s3d0,  b ,
                                  args=(k,eta,1.0),   epsrel=erl, epsabs=0.0)
        # Add all the four parts of val1 and repeat for val2
        val1[idx]  =  val1A + val1B + val1C + val1D
        val2[idx]  =  val2A + val2B + val2C + val2D
        idx  =  idx + 1  # incrementing the array index
    print("%.1f , 0 , %3.16e , %3.16e , %3.16e"% ( k,
                                                  val1[0],val1[200],val1[-1]))
    print("%.1f , 1 , %3.16e , %3.16e , %3.16e"% ( k,
                                                  val2[0],val2[200],val2[-1]))
    plt.semilogy(eta_array, val1, ls = "-", color = clr, label=
                                              r"$F_{%.1f}(\beta = 0)$"%k )
    plt.semilogy(eta_array, val2,  ls = "--", color = clr, label=
                                              r"$F_{%.1f}(\beta = 1)$"%k )
plt.grid();  plt.legend(loc="best",frameon=False)
plt.xlabel(r"Fermi Energy fraction, $\eta$=$E_F/kT$")
plt.ylabel(r"$FD_k(\eta,\beta)$")
plt.show()
#End of Table A.2
```