Ionisationsgleichgewicht von Helium

Manfred Hanke (Gruppe 5: Hanke / Nieva)

6. Dez 2006

1 Zielsetzung

Es soll der Ioniationsgrad $x = \frac{N_\mathrm{E}}{N_\mathrm{He}}$ (Zahl der freien Elektronen pro Helium-Teilchen) eines Helium-Plasmas vorgegebener Temperatur T und Anzahldichte n_He berechnet werden.

2 Methoden

Das thermodynamische Gleichgewicht wird von den Saha-Gleichungen beschrieben:

$$n_E \cdot \frac{n_\mathrm{II}}{n_\mathrm{I}} = c_1 \quad \text{und} \quad n_E \cdot \frac{n_\mathrm{III}}{n_\mathrm{II}} = c_2 \qquad \text{mit} \quad c_i = \frac{g_{i,\mathrm{up}}}{g_{i,\mathrm{low}}} \cdot 2 \cdot \left(\frac{2\pi mkT}{h^2}\right)^{\frac{3}{2}} \cdot \exp\left(\frac{-E_i}{kT}\right)$$

Mit $E_i = \frac{hc}{\lambda_i}$ und $C_i := \frac{n_\mathrm{He}}{c_i} = 2.07{\cdot}10^{-16} \cdot \frac{g_{i,\mathrm{low}}}{g_{i,\mathrm{up}}} \cdot \frac{n_\mathrm{He}}{\mathrm{cm}^{-3}} \cdot \left(\frac{T}{\mathrm{K}}\right)^{-\frac{3}{2}} \cdot \exp\left(\frac{1.44{\cdot}10^8}{(\lambda_i/\mathrm{A}) \cdot (T/\mathrm{K})}\right)$ folgt:

$$n_\mathrm{I} = n_\mathrm{II} \cdot C_1 \cdot x, \qquad n_\mathrm{II} = n_\mathrm{III} \cdot C_2 \cdot x \ .$$

Setzt man das in

$$x \cdot (n_\mathrm{I} + n_\mathrm{II} + n_\mathrm{III}) = x \cdot n_\mathrm{He} = n_\mathrm{E} = n_\mathrm{II} + 2n_\mathrm{III}$$

ein, erhält man:

$$x \cdot ([n_\mathrm{III} \cdot C_2 \cdot x] \cdot C_1 \cdot x + n_\mathrm{III} \cdot C_2 \cdot x + n_\mathrm{III}) = n_\mathrm{III} \cdot C_2 \cdot x + 2n_\mathrm{III} \qquad |: n_\mathrm{III}$$

$$\Leftrightarrow \qquad C_1 C_2 x^3 + C_2 x^2 + (1 - C_2)x - 2 = 0 \tag{1}$$

Damit bestimmt sich $x(T, n_\mathrm{He})$ als Lösung dieser kubischen Gleichung.

Da nur $0 \leq x \leq 2$ physikalisch sinnvoll ist (Jedes Helium-Atom kann maximal 2 Elektronen abgeben.), kann man die Nullstelle von (1) sicher mit dem Bisektionsverfahren bestimmen.

3 Durchführung

Die vorherige `SUBROUTINE Bisect` ist in eine `FUNCTION` umgewandelt worden. Die Funktion, deren Nullstellen gesucht werden soll, ist eine allgemeine `FUNCTION cubic` zur Berechnung eines kubischen Polynoms, dessen Koeffizienten in einem `COMMON`-Block übergeben werden.

Die oben diskutierte Methode zur Berechnung des Ionisationsgrades wird in der `FUNCTION IonDeg` umgesetzt, die als Eingabe-Parameter T und n_He erwartet. Die Berechnung der der Exponentialfunktionen $\exp\left(\frac{1.4410^8}{(\lambda_i/\mathrm{A})\cdot(T/\mathrm{K})}\right)$ für die Konstanten C_1 und C_2 erfolgt mit `DOUBLE PRECISION`, da sie dann für einen größeren Temperaturbereich endliche Werte liefert. (Unterhalb 1 242.367 K wird $C_1 \cdot C_2 > 1.8{\cdot}10^{308}$.) Anschließend werden die Koeffizienten des kubischen Polynoms initialisiert und schließlich der Ionisationsgrad mit der Bisektions-Methode und den Randwerten 0 und 2 berechnet.
Im Hauptprogramm werden dann zwei Wertetabellen für die Dichten $n_\mathrm{He} = 10^{14}/\mathrm{cm}^3$ und $n_\mathrm{He} = 10^{18}/\mathrm{cm}^3$ und Temperaturen bis 120 000 K erstellt und in separate Dateien geschrieben, die dann mit einem einfachen Gnuplot-Script graphisch dargestellt werden können.

4 Programm

4.1 Fortran-Code

```
      REAL FUNCTION Bisect(y, x10, x20)
c      zero-finding by sisection
      IMPLICIT NONE
      REAL y, x10, x20, x1, x2, x
      x1 = x10
      x2 = x20
      IF(y(x1)*y(x2) .GT. 0.0) THEN
        WRITE(*,*) "ERROR: bad choice of x1 and x2"
        RETURN
      ENDIF
10    x = 0.5*(x1+x2)
      IF(y(x1)*y(x) .LE. 0.0) THEN
        x2 = x
      ELSE
        x1 = x
      ENDIF
      IF(ABS(x1-x2) .GT. 1e-6) GOTO 10
      Bisect = x1
      END
c
```

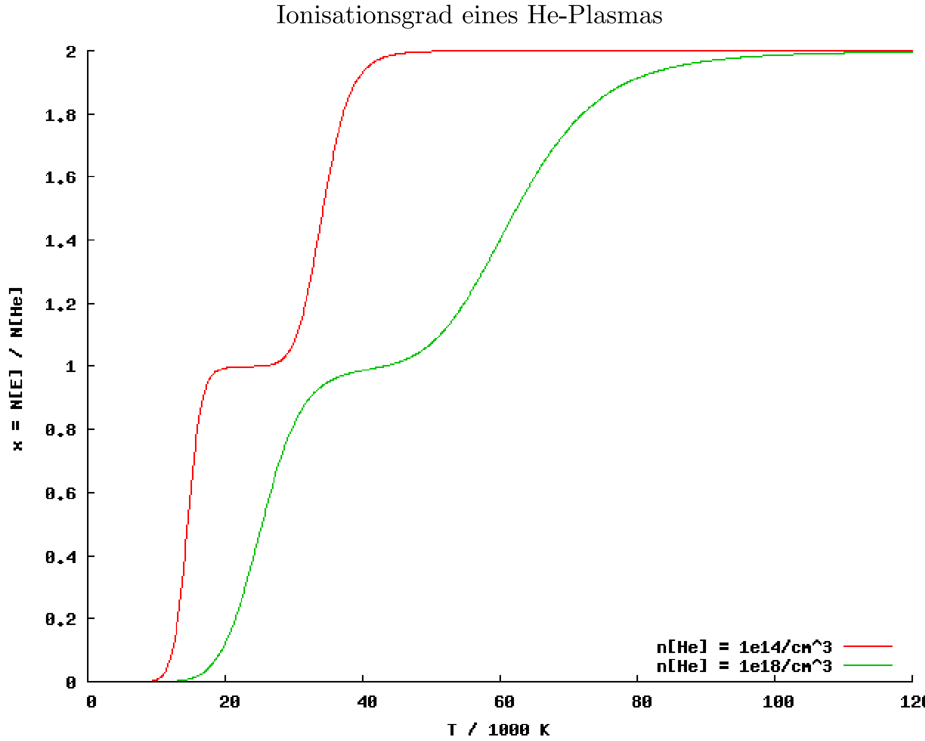
```
      REAL FUNCTION cubic(x)
c      evaluates a cubic polynom with common coefficients
      REAL x
      DOUBLE PRECISION cubic3, cubic2, cubic1, cubic0
      COMMON cubic3, cubic2, cubic1, cubic0
      cubic = ((cubic3*x + cubic2)*x + cubic1)*x + cubic0
      END
c
      REAL FUNCTION IonDeg(T, nHe)
      REAL T, nHe
      DOUBLE PRECISION C1, C2, cubic3, cubic2, cubic1, cubic0
      COMMON cubic3, cubic2, cubic1, cubic0
      EXTERNAL cubic
      REAL cubic, Bisect
      C1 = 1.035e-16 * nHe * (T**(-1.5)) * exp(DBLE(2.86e5/T))
      C2 = 4.140e-16 * nHe * (T**(-1.5)) * exp(DBLE(6.32e5/T))
      cubic3 = C1*C2
      cubic2 = C2
      cubic1 = 1.D0 - C2
      cubic0 = -2.D0
      WRITE(*,*) cubic3
      IonDeg = Bisect(cubic, 0.0, 2.0)
      END
c
      PROGRAM Ionization
      IMPLICIT NONE
      REAL nHe, T, IonDeg
c
      nHe = 1.0e14
      T = 1300.0
      OPEN(1, FILE='1b_Ionization_nHe1e14.dat')
100   WRITE(*,'(F8.0,F10.6)') T, IonDeg(T, nHe)
      T = T + 100.0
      IF(T .LE. 120000.0) GOTO 100
      CLOSE(1)
      STOP
c
      nHe = 1.0e18
      T = 1300.0
      OPEN(1, FILE='1b_Ionization_nHe1e18.dat')
200   WRITE(1,'(F8.0,F10.6)') T, IonDeg(T, nHe)
      T = T + 100.0
      IF(T .LE. 120000.0) GOTO 200
      CLOSE(1)
      END
```

4.2 Gnuplot-Script

```
set border 3
set xtics nomirror
set ytics nomirror
set term png

set output "1b_Ionization.png"
set xlabel "T / 1000 K"
set ylabel "x = N[E] / N[He]"
set key bottom
plot "1b_Ionization_nHe1e14.dat" u ($1/1000):2 w l t "n[He] = 1e14/cm^
     "1b_Ionization_nHe1e18.dat" u ($1/1000):2 w l t "n[He] = 1e18/cm^
```

5 Ergebnisse



Man sieht deutlich, daß sich das dünnere Heliumgas schon bei geringeren Temperaturen ionisieren läßt als das dichtere und daß Schwellen beim Ionisationsgrad $x = 1$ auftreten. Die Schwelle ist beim dünneren Plasma deutlicher ausgeprägt.

| Mh