```
(* Radiative Quenching, Chapter 3 *)
(* This one uses my paper *)

(*electron mass *)
me = 1;
(* Proton Mass *)
mp = 1836;
(* Reduced Mass *)
μ = mp/2;
hbar = 1;
hbarSI = 1.054571817*^-34;
cLight = 137.036 ;(*SpeedOfLight, atomic units*)
cLightSI = 299 792 458 ;(*SpeedOfLight, m/s*)
oneSec = 2.418884326509*^-17; (* 1 sec SI ↔ 1 sec AU *)
aBorh = 5.29177*^-11;  (* Si ↔ AU *)

(* RawData format R,E, E + 1/R,A,p *)

vSg1RawData =
  Import["/Users/zelimir/Work/Physics-Thesis/thesis-2/ZelimirOverleaf/
     MathematicaCodeFinal/gerade1sV2.mx"];
vSu2RawData =
  Import["/Users/zelimir/Work/Physics-Thesis/thesis-2/ZelimirOverleaf/
     MathematicaCodeFinal/ungerade1sV2.mx"];

vSg1Data = vSg1RawData⟦1 ;; 44⟧;
vSu2Data = vSu2RawData⟦1 ;; 44⟧;

(* Calculate the Transition Dipole Moment *)
(* Upper Limit for integration *)
maxR = vSg1Data⟦Length[vSg1Data]⟧⟦1⟧;
Print["maxR=", maxR]

(* Load Potential curves for 1sg, 2sg states
"R","p", "A","E","E+1/r"*)
(* 1sg state, potential curve *)
vG = Interpolation[
   Transpose[{vSg1Data⟦All, 1⟧, vSg1Data⟦All, 3⟧}], InterpolationOrder → 3];
(* 2sg state, potential curve *)
vU = Interpolation[
   Transpose[{vSu2Data⟦All, 1⟧, vSu2Data⟦All, 3⟧}], InterpolationOrder → 3];
```

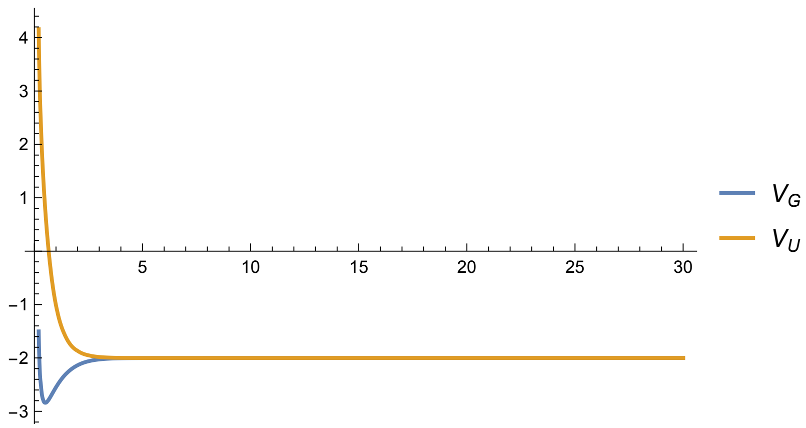
```
deltaV[r_] := Abs[vG[r] - vU[r]];
deltaV[20]
Plot[{vG[r], vU[r]}, {r, 0.2, maxR}, PlotRange → Full, PlotLegends → {"V_G", "V_U"}]
```

maxR=30

Out[∘]=

3.64958×10^{-9}

Out[∘]=



In[∘]:=

```
Print[vSg1Data[[19]][[1]], ", ", vSg1Data[[19]][[4]], ", ",
 vSg1Data[[19]][[5]], ", ", vSu2Data[[19]][[4]], ", ", vSu2Data[[19]][[5]]]
Print[vSg1Data[[20]][[1]], ", ", vSg1Data[[20]][[4]], ", ",
 vSg1Data[[20]][[5]], ", ", vSu2Data[[20]][[4]], ", ", vSu2Data[[20]][[5]]]
```

5., 3437.09, 5.24591, 17 397.4, 5.2446

6, 127 499., 6.24594, 54 879.4, 6.2457

In[ ]:=

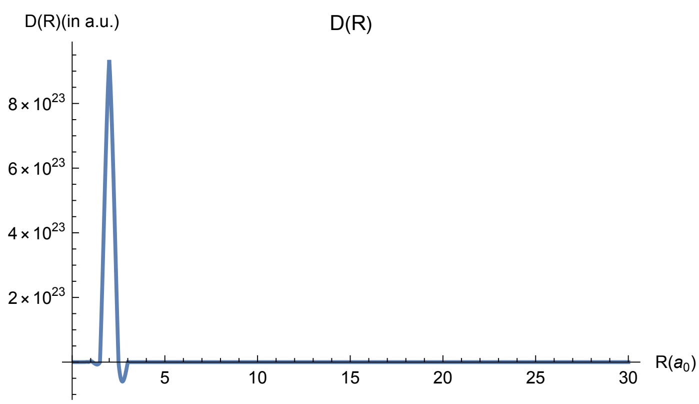
```
(* Compute the Dipole momement D(r) for the single value of R *)
singleDR[R_, a1_, p1_, a2_, p2_, maxDistance_] :=
  Module[{lSol1, mSol1, lSol2, mSol2, norm1, norm2, dInt, lm1, lm2},
   (* 1s wavefunction *)
   (* x = λ, y = μ **)
   l1 = NDSolveValue[{(x^2 - 1) L''[x] + x * L'[x] + (a1 + 2 R x - p1^2 x^2) L[x] == 0 ,
      L[1.01] == 0, L'[1.01] == 1 }, L, {x, 1.01, maxDistance}];
   m1 = NDSolveValue[{(1 - y^2) M''[y] - y * M'[y] - (a1 + p1^2 y^2) M[y] == 0,
      M[-0.999] == 0, M'[-0.999] == 1}, M, {y, -.999, .999}];
   (* 2s wavefunction *)
   l2 = NDSolveValue[{(x^2 - 1) L''[x] + x * L'[x] + (a2 + 2 R x - p2^2 x^2) L[x] == 0 ,
      L[1.01] == 0, L'[1.01] == 1 }, L, {x, 1.01, maxDistance}];
   m2 = NDSolveValue[{(1 - y^2) M''[y] - y * M'[y] - (a2 + p2^2 y^2) M[y] == 0,
      M[-0.999] == 0, M'[-0.999] == 1}, M, {y, -.999, .999}];

   Print["R=", R];
   (* Compute <ψ|ψ> = <ψ1s|ψ1s> + <ψ2s|ψ2s> *)
   (* <ψ1s|ψ1s> *)
   norm1 =
    Sqrt[NIntegrate[Abs[l1[x] × m1[y]]^2, {y, -.9, .9}, {x, 1.01, maxDistance}]];
   (* <ψ2s|ψ2s> *)
   norm2 =
    Sqrt[NIntegrate[Abs[l2[x] × m2[y]]^2, {y, -.9, .9}, {x, 1.01, maxDistance}]];

   lmNorm1[x_, y_] := 1/norm1 l1[x] × m1[y];
   lmNorm2[x_, y_] := 1/norm2 l2[x] × m2[y];

   dInt = (R/2)^3 NIntegrate[Conjugate[lmNorm1[x, y]] x y lmNorm2[x, y] (x^2 - y^2),
      {y, -.9, .9}, {x, 1.001, maxDistance}];
   {R, dInt }
  ];


(*singleDR[vSg1Data[[19]][[1]],vSg1Data[[19]][[4]],
  vSg1Data[[19]][[5]],vSu2Data[[19]][[4]],vSu2Data[[19]][[5]],30]
 singleDR[vSg1Data[[20]][[1]],vSg1Data[[20]][[4]],
  vSg1Data[[20]][[5]],vSu2Data[[20]][[4]],vSu2Data[[20]][[5]],30]*)
```

In[ ]:=

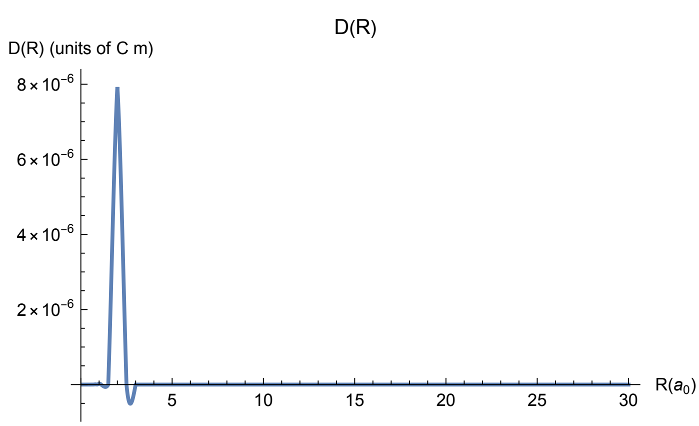
```
(* Get all Rs, and coefficients A and p and calculate D(R) for all Rs *)
(* RawData format R,E, E + 1/R,A,p *)
inputData = Transpose[{vSg1Data[[All, 1]], vSg1Data[[All, 4]],
    vSg1Data[[All, 5]], vSu2Data[[All, 4]], vSu2Data[[All, 5]]}];

allDR = Map[singleDR[#[[1]], #[[2]], #[[3]], #[[4]], #[[5]], maxR] &, inputData];
```

```
(* Dipole moment in SI units *)
drAUtoSI = 8.478*^-30; (* Cm *)
allDRSI = allDR;
allDRSI[[All, 2]] = allDR[[All, 2]] * drAUtoSI;
allDRSI

dR = Interpolation[
   Transpose[{allDR[[All, 1]], allDR[[All, 2]]}], InterpolationOrder → 3];
Plot[dR[r], {r, 0.1, maxR}, PlotLabel → "D(R)",
 AxesLabel → {"R(a0)", "D(R)(in a.u.)"}, PlotRange → Full]

dRSI = Interpolation[
   Transpose[{allDRSI[[All, 1]], allDRSI[[All, 2]]}], InterpolationOrder → 3];
Plot[dRSI[r], {r, 0.1, maxR}, PlotLabel → "D(R)",
 AxesLabel → {"R(a0)", "D(R)  (units of C m)"}, PlotRange → Full]
```

Out[ ]=

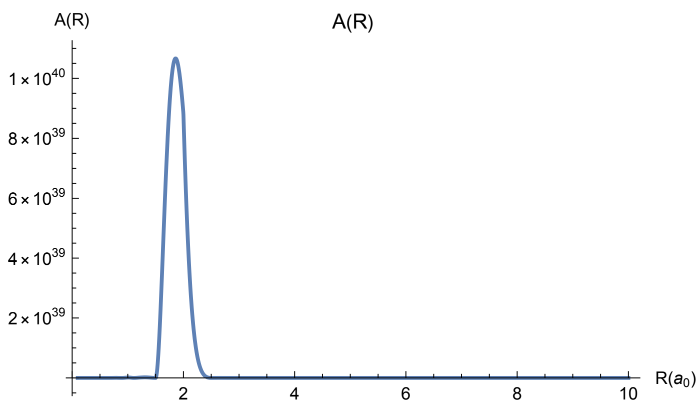


Out[ ]=



```
aR[R_] := 4/(3 hbar cLight^3) (dR[R])^2 ((vU[R] - vG[R])/hbar)^3;

(* From a.u. to SI units *)
aRAUtoSI = 4.1341*^16;
aRSI[R_] := aR[R] * aRAUtoSI;


Plot[aR[r], {r, .1,  10}, PlotLabel → "A(R)",
 AxesLabel → {"R(a0)", "A(R)"}, PlotRange → Full]
Plot[aRSI[r], {r, .1,  10}, PlotLabel → "A(R)",
 AxesLabel → {"R(a0)", "A(R)  (s^-1)"}, PlotRange → Full]
```

Out[ ]=



Out[ ]=