Potential energy surfaces for singlet and triplet states of the LiH_2^+ system and quasi-classical trajectory study on the H + LiH^+ and H^+ + LiH reactions

Javier Hernández,[1,2] Cristina Sanz,[1] Pedro Alberto Enríquez,[3]

Miguel González[4,*] and Miguel Paniagua[1,*]

[1] Dept. de Química Física Aplicada, Univ. Autónoma de Madrid, Cantoblanco, Spain

[2] Depto. de Química Física, Univ. de Salamanca, Spain

[3] Dept. de Química, Univ. de La Rioja, Logroño, Spain

[4] Dept. de Ciència de Materials i Química Física and IQTC, Univ. de Barcelona, Barcelona, Spain

Supplementary Information

Table S1. Exoergicity of $LiH^+(X^2\Sigma^+) + H(^2S) \rightarrow Li^+(^1S) + H_2(X^1\Sigma_g^+)$ and $LiH(X^1\Sigma^+) + H^+ \rightarrow Li(^2S) + H_2^+(X^2\Sigma_g^+)$. Comparison with the previous results. Page 4.

Table S2. Stationary point [T-shape (C_{2v}) minimum] of the $1^1A'$ ground PES. Comparison with the previous results. Page 4.

Table S3. Stationary point [Li-H---H collinear ($C_{\infty v}$) minimum 1] of the $2^1A'$ excited PES. Comparison with the previous results. Page 5.

Table S4. Stationary point [Li----H-H collinear ($C_{\infty v}$) minimum 2] of the $2^1A'$ excited PES. Comparison with the previous results. Page 5.

Table S5. Stationary point [Li---H---H collinear ($C_{\infty v}$) transition state] of the $2^1A'$ excited PES. Comparison with the previous results. Page 6.

Figure S1. Equipotential contour plots of the ground $1^1A'$ PES as a function of the internal distances, r_{HH} and r_{LiH} and the angle between them indicated at the top of each panel. The zero energy corresponds to the separated atoms/ions limit. Page 7.

Figure S2. Same as in Figure 1 but for the excited $2^1A'$ PES. Page 8.

Figure S3. Same as in Figure 1 but for the excited $1^1A''$ PES. Page 9.

Figure S4. Same as in Figure 1 but for the excited $1^3A'$ PES. Page 10.

Figure S5. Same as in Figure 1 but for the excited $2^3A'$ PES. Page 11.

Figure S6. Same as in Figure 1 but for the excited $1^3A''$ PES. Page 12.

Figure S7. Cross sections for the reactions on the ground $1^1A'$ PES. Page 13.

Figure S8. Cross sections for the reactions on the excited $2^1A'$ PES. Page 14.

Figure S9. Cross sections for the reactions on the excited $1^1A''$ PES. Page 15.

Figure S10. Cross sections for the reactions on the excited $1^3A'$ PES. Page 16.

Figure S11. Cross sections for the reactions on the excited $1^3A''$ PES. Page 17.

Figure S12. Schematic representation of the minimum energy paths for the exchange reaction channel of the six PESs considered. Page 18.

Figure S13. A more detailed view of the minimum energy paths for the exchange reaction channel of the $1^1A'$ and $1^3A'$ PESs. Page 19.

References. Page 20.

Fortran code 1. Analytical expression of the ground $1^1A'$ PES. Page 21.

Fortran code 2. Analytical expression of the excited $2^1A'$ PES. Page 26.

Fortran code 3. Analytical expression of the excited $1^1A''$ PES. Page 33.

Fortran code 4. Analytical expression of the excited $1^3A'$ PES. Page 38.

Fortran code 5. Analytical expression of the excited $2^3A'$ PES. Page 44.

Fortran code 6. Analytical expression of the excited $1^3A''$ PES. Page 49.

Table S1. Exoergicity of $LiH^+(X^2\Sigma^+)$ + $H(^2S)$ → $Li^+(^1S)$ + $H_2(X^1\Sigma_g^+)$ (reverse of reaction (1)) and $LiH(X^1\Sigma^+)$ + H^+ → $Li(^2S)$ + $H_2^+(X^2\Sigma_g^+)$ (reverse of reaction (2)), that take place on the $1^1A'$ and $2^1A''$ PESs, respectively. Comparison with the previous results.

Result	Exoergicity reverse of react. (1) (eV)	Exoergicity reverse of react. (2) (eV)
MRVB,[a] MRCI (Ref. 1)	4.6091	0.27230
MRCI (Ref. 2)	4.5883	0.299
MRCI (Ref. 3)	4.5998	0.20220[b]
MRCI Present work	4.608	0.274

[a] Multireference Valence Bond.
[b] The lower value observed for ref. 3 arises from the larger value obtained for the D_e of $LiH(X^1\Sigma^+)$, that is 0.0744 eV larger than the experimental value.

Table S2. Stationary point [T-shape (C_{2v}) minimum] of the $1^1A'$ ground potential energy surface. Comparison with the previous results (the energy is given relative to $Li^+(^1S)$ + $H_2(X^1\Sigma_g^+)$).[a]

Result	R_{HH} (Å)	R_{LiH} (Å)	$(H\text{-}Li\text{-}H)^+$ angle (deg)	Energy (eV)
CI (Ref. 4)	0.751	2.0426	21.2	-0.258
MRVB, MRCI (Ref. 1)	0.7514	1.952	22.19	-0.286
MRCI (Ref. 2)	0.7503	2.0221	21.38	-0.267
MRCI (Ref. 3)	0.751	2.0271	21.35	-0.266
MRCI Present work	0.746	2.045	21.02	-0.268

[a]This minimum is located in the $Li^+(^1S)$ + $H_2(X^1\Sigma_g^+)$ region of the PES.

Table S3. Stationary point [Li-H---H collinear ($C_{\infty v}$) minimum 1] of the $2^1A'$ excited potential energy surface. Comparison with the previous results (the energy is given relative to $LiH(X^1\Sigma^+) + H^+$).[a]

Result	R_{HH} (Å)	R_{LiH} (Å)	Energy (eV)
MRVB, MRCI (Ref. 1)	2.498	1.656	-1.315
MRCI (Ref. 2)	2.517	1.645	-1.290
MRCI (Ref. 3)	2.517	1.615	-1.292
MRCI Present work	2.513	1.646	-1.282

[a]This minimum is located in the $LiH(X^1\Sigma^+) + H^+$ region of the PES.

Table S4. Stationary point [Li----H-H collinear ($C_{\infty v}$) minimum 2] of the $2^1A'$ excited potential energy surface. Comparison with the previous results (the energy is given relative to $Li(^2S) + H_2^+(X^2\Sigma_g^+)$).[a]

Result	R_{HH} (Å)	R_{LiH} (Å)	Energy (eV)
MRVB, MRCI (Ref. 1)	1.006	3.695	-0.585
MRCI (Ref. 2)	1.062	3.529	-0.534
MRCI (Ref. 3)	1.064	3.494	-0.547
MRCI Present work	1.065	3.559	-0.532

[a]This minimum is located in the $Li(^2S) + H_2^+(X^2\Sigma_g^+)$ region of the PES.

Table S5. Stationary point [Li---H---H collinear ($C_{\infty v}$) transition state] of the $2^1A'$ excited potential energy surface. Comparison with the previous results (the energy is given relative to the Li-H---H collinear ($C_{\infty v}$) minimum 1).

Result	R_{HH} (Å)	R_{LiH} (Å)	Energy (eV)
MRVB, MRCI (Ref. 1)	1.600	2.588	1.088
MRCI (Ref. 2)	1.597	2.552	1.057
MRCI (Ref. 3)	1.614	2.558	1.104
MRCI Present work	1.627	2.591	1.077

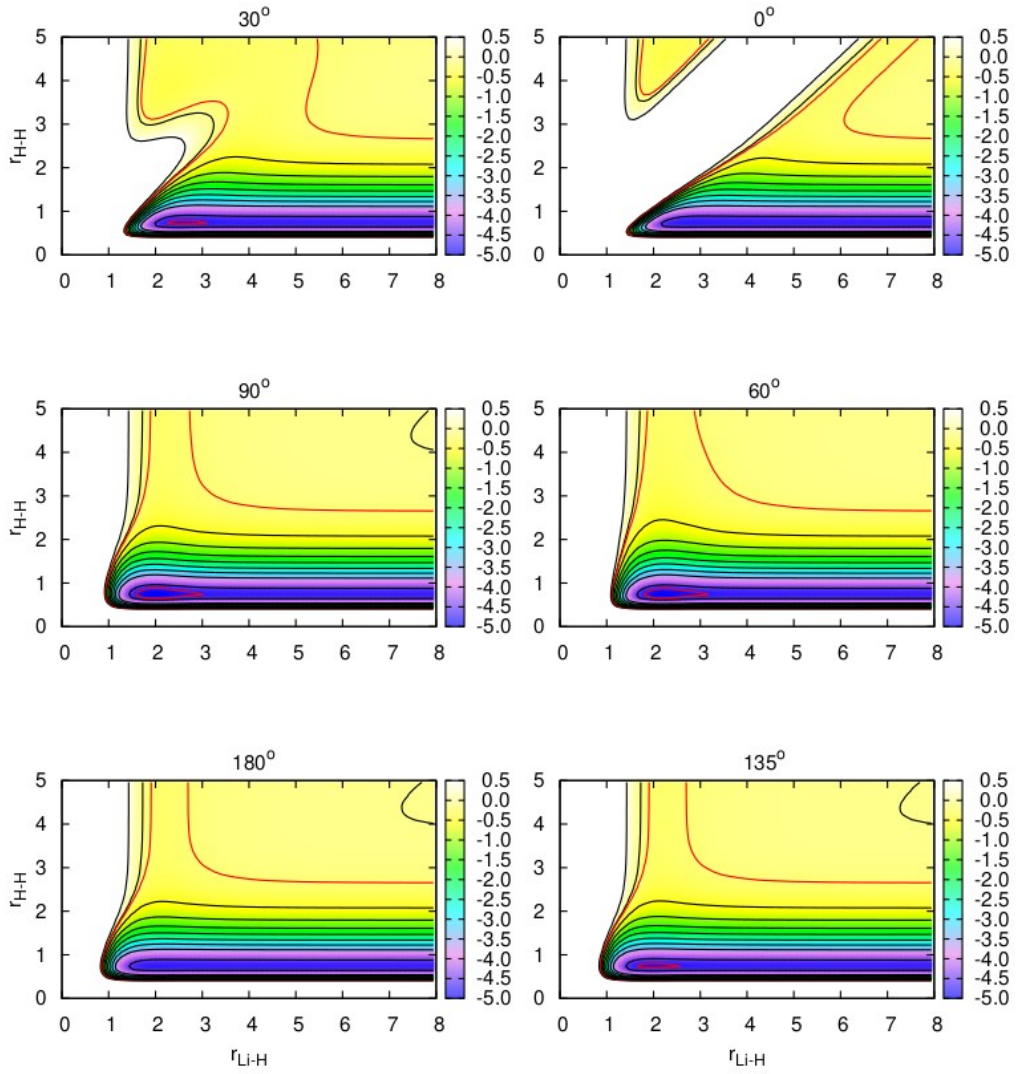

Figure S1. Equipotential contour plots of the ground $1^1A'$ PES as a function of the internal distances, r_{HH} and r_{LiH} and the angle between them indicated at the top of each panel. The contour energies are indicated at the right of each panel in eV, zero energy, 0 eV, corresponds to the separated atoms/ions limit in their ground states [$Li^+(^1S)$, $H(^2S)$, $H(^2S)$], and all distances are in Å. The contours are spaced by 0.5 eV, besides, the contour levels -0.1 eV and -4.8 eV (in red color) are also included for comparison with Figure 1 of ref. 3.

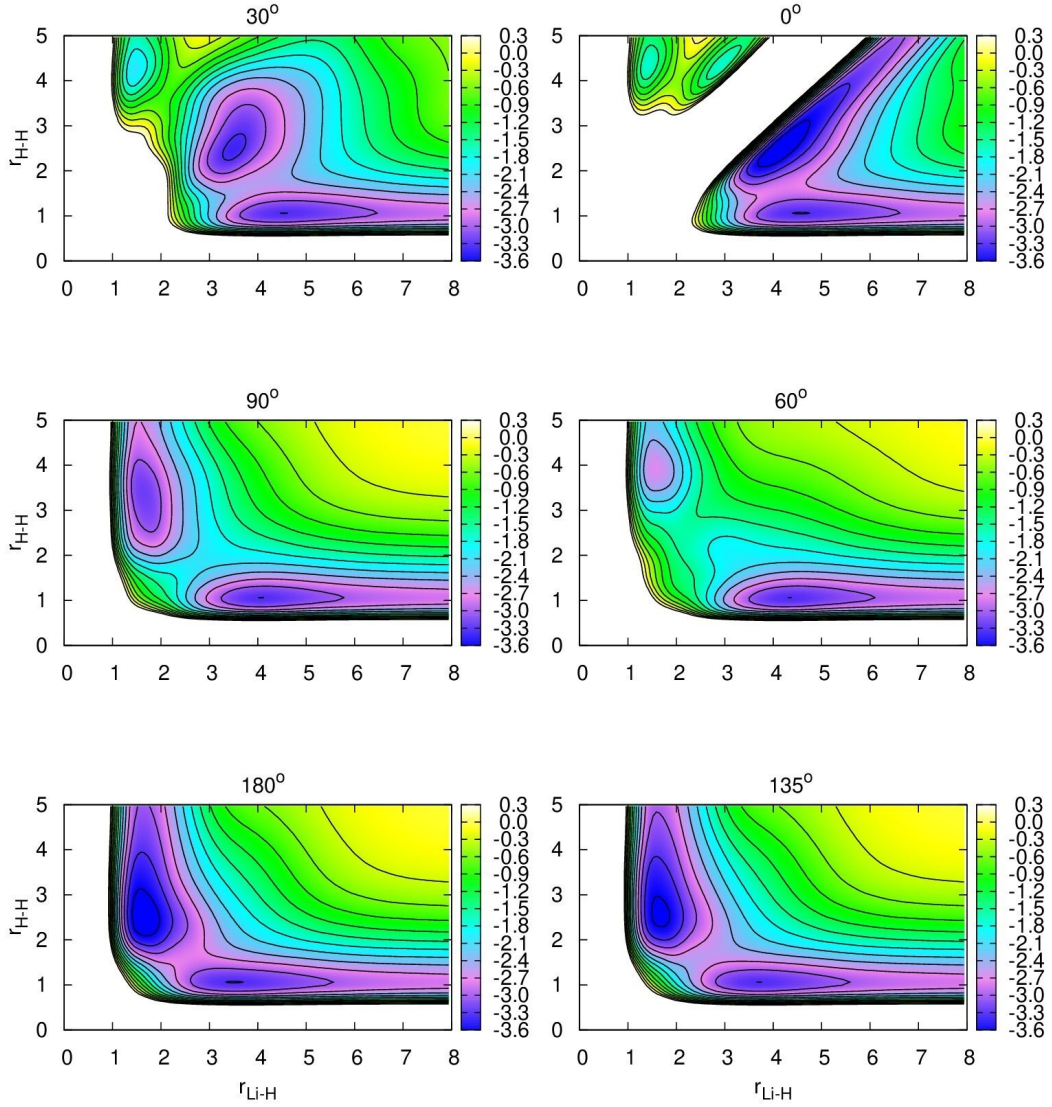

Figure S2. Same comments as in Figure S1 but for the excited $2^1A'$ PES. Here the zero energy corresponds to the separated atoms/ions limit: Li(^{2}S), H(^{2}S), H^+. The contours are spaced by 0.3 eV, besides, the contour level -3.0 eV (in red color) is also included for comparison with Figure 2 of ref. 3.

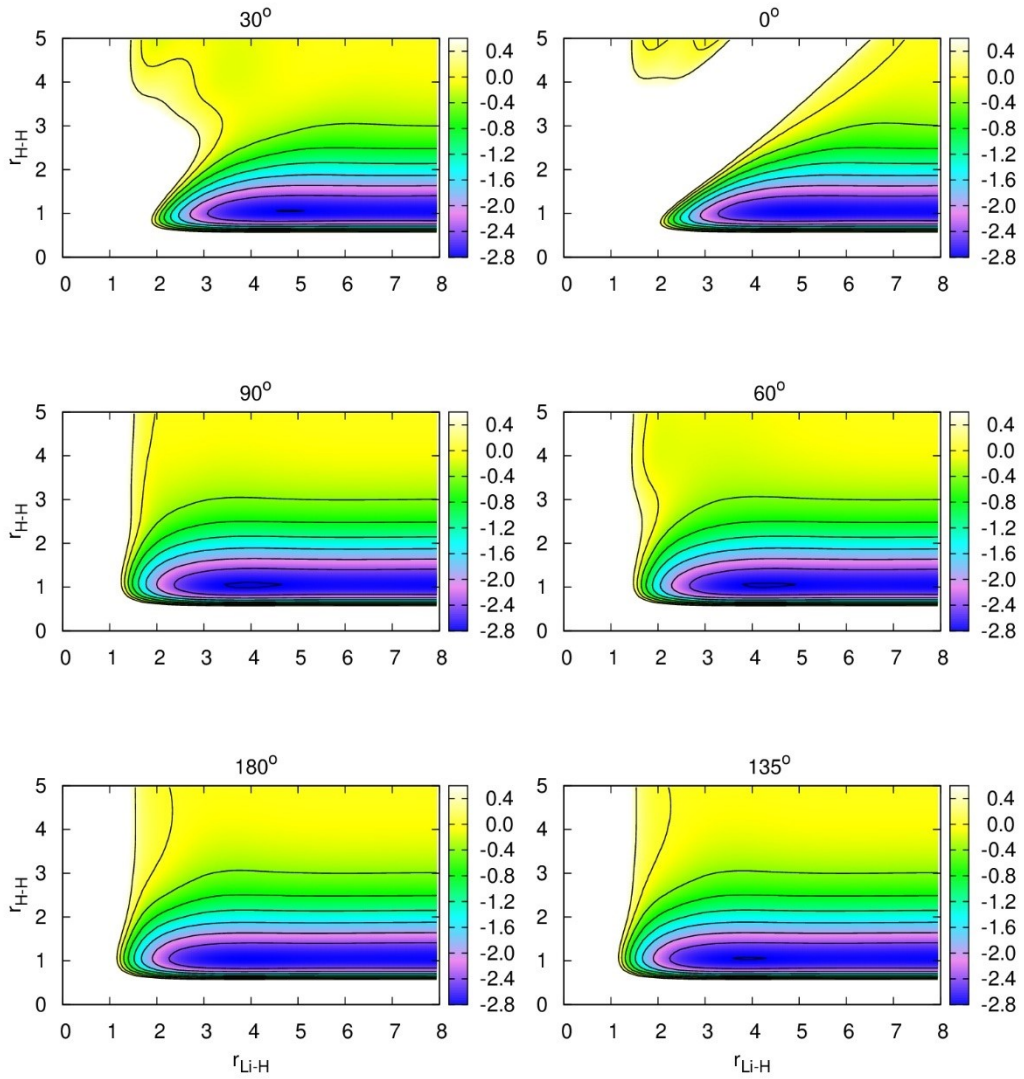

Figure S3. Same comments as in Figure S1 but for the excited $1^1A''$ PES. Here the zero energy corresponds to the separated atoms/ions limit: Li(^{2}P), H(^{2}S), H^+.

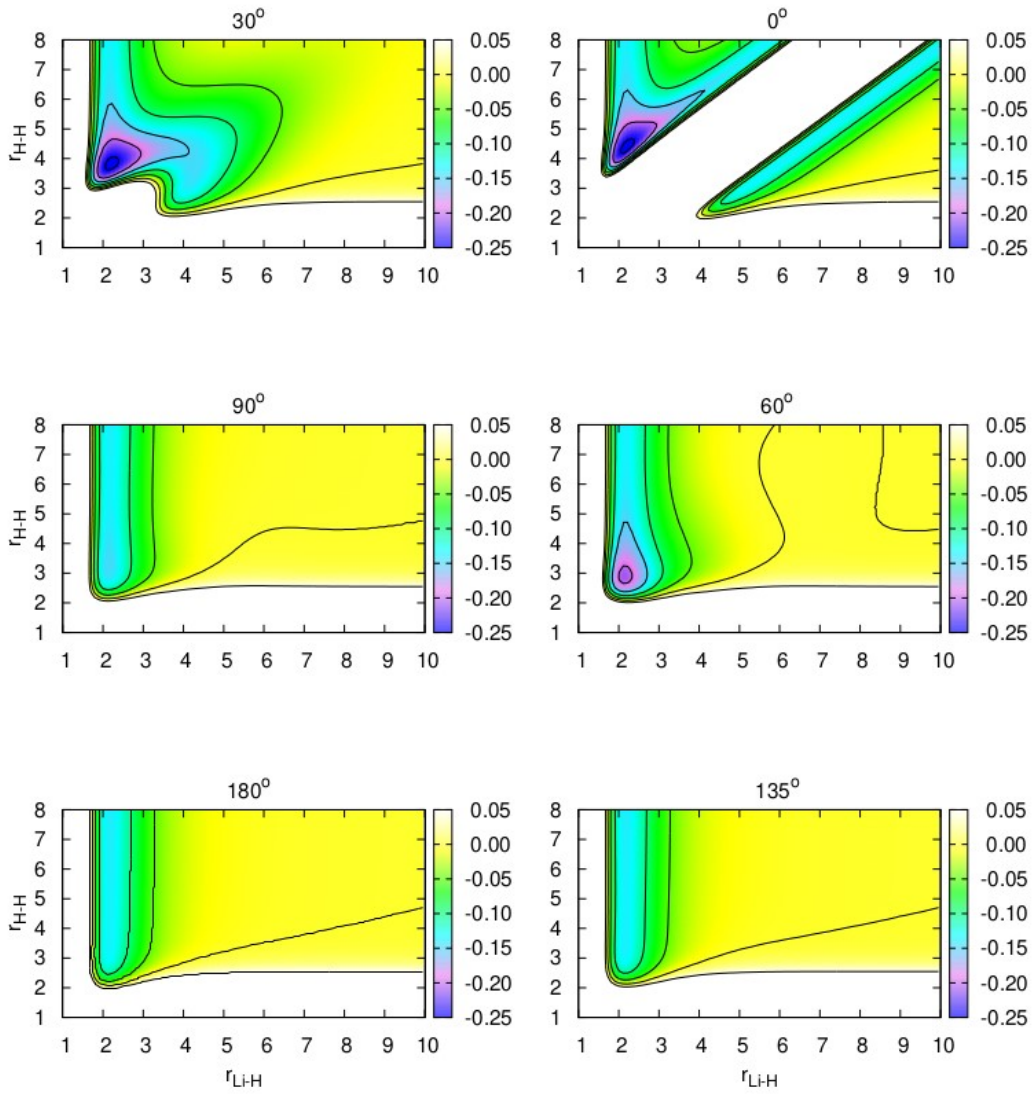

Figure S4. Same comments as in Figure S1 but for the excited $1^3A'$ PES. Here, as for the ground state $1^1A'$, the zero energy corresponds to the separated atoms/ions limit in their ground states [$Li^+(^1S)$, $H(^2S)$, $H(^2S)$].

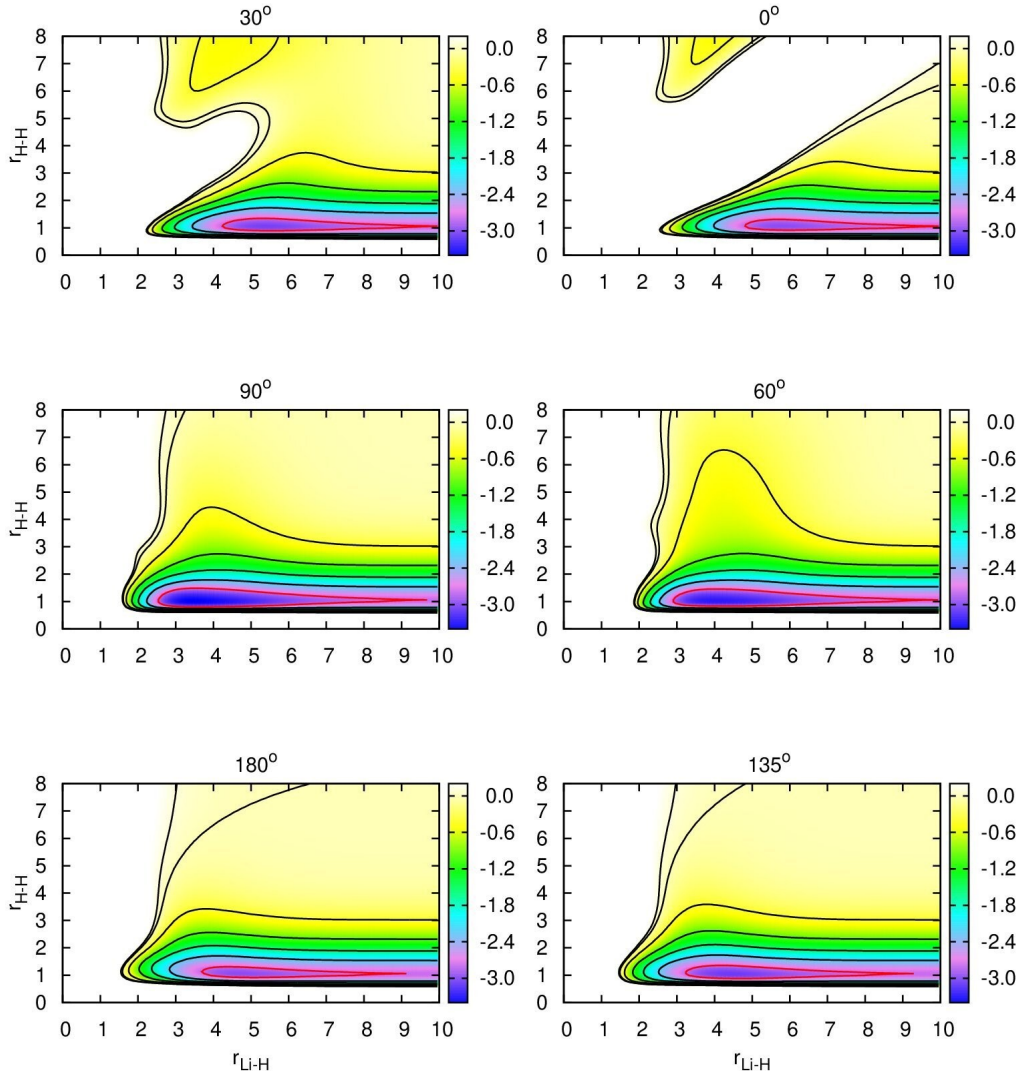

Figure S5. Same comments as in Figure S1 but for the excited $2^3A'$ PES. Here, as for the excited state $2^1A'$, the zero energy corresponds to the separated atoms/ions limit: $Li(^2S)$, $H(^2S)$, H^+.

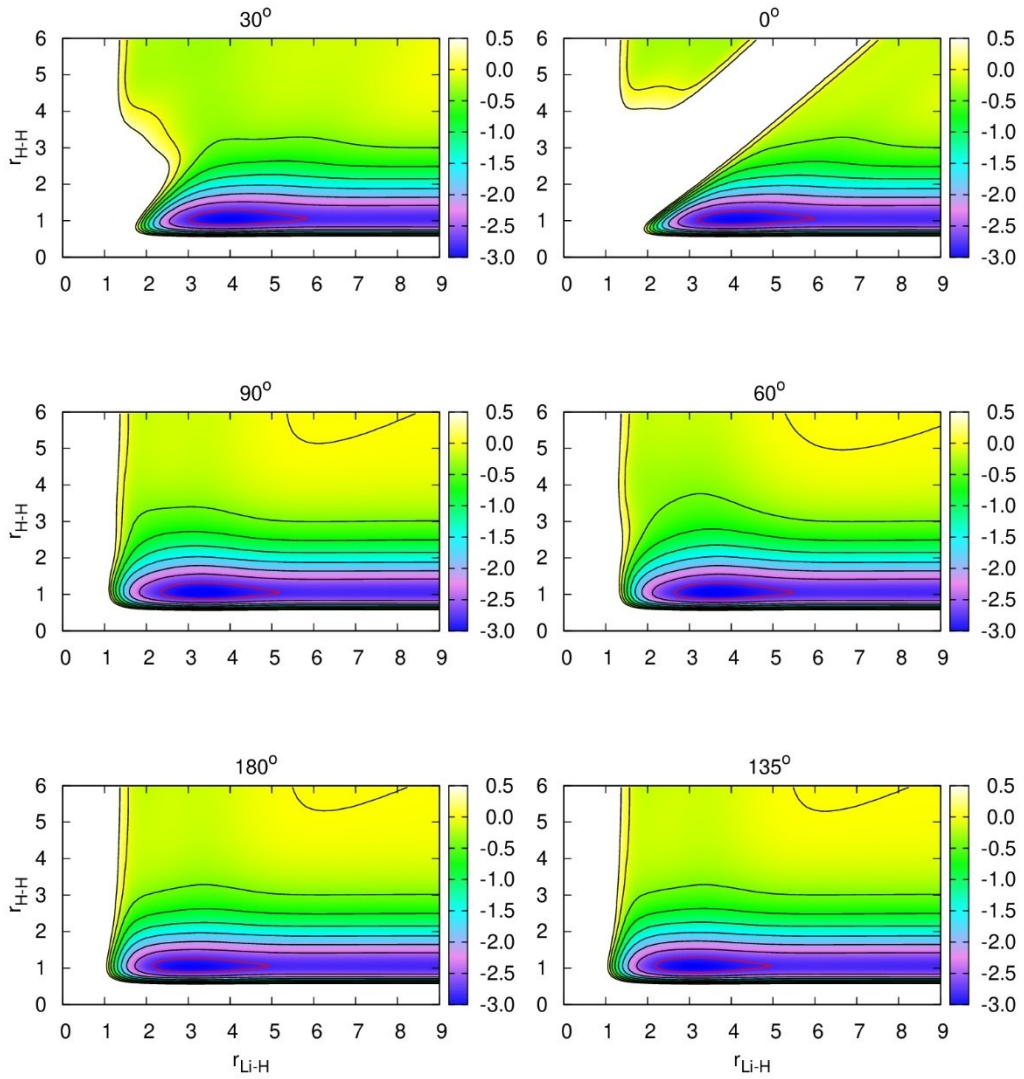

Figure S6. Same comments as in Figure S1 but for the excited $1^3A''$ PES. Here, as for the excited state $1^1A''$, the zero energy corresponds to the separated atoms/ions limit: $Li(^2P)$, $H(^2S)$, H^+.

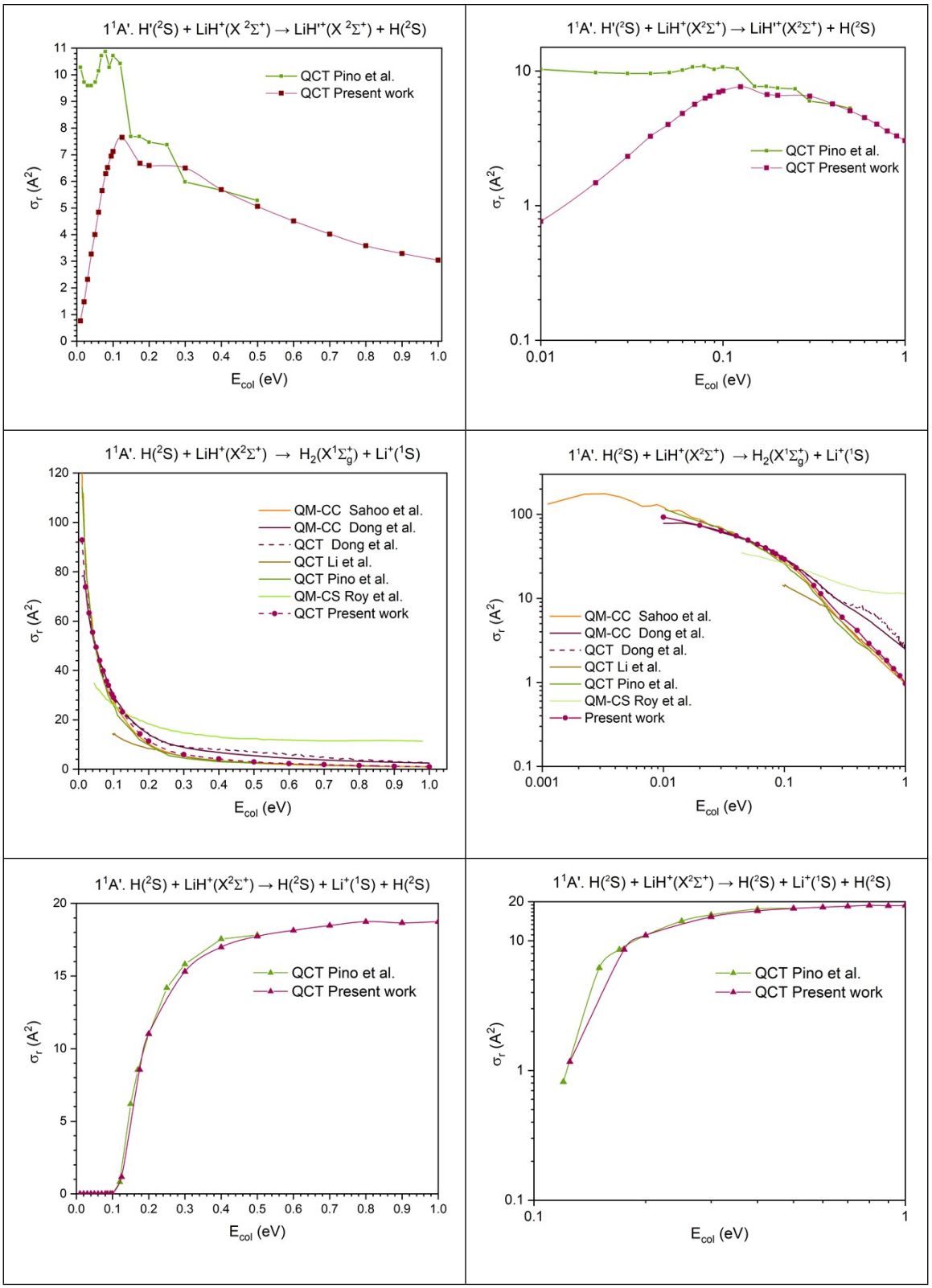

Figure S7. Cross sections for the reactions on the ground 1^1A' PES and comparison with other theoretical results (refs. 5-9): exchange (top), H atom transfer (middle) and collision induced dissociation (bottom). The double logarithmic representation is given on the right column.

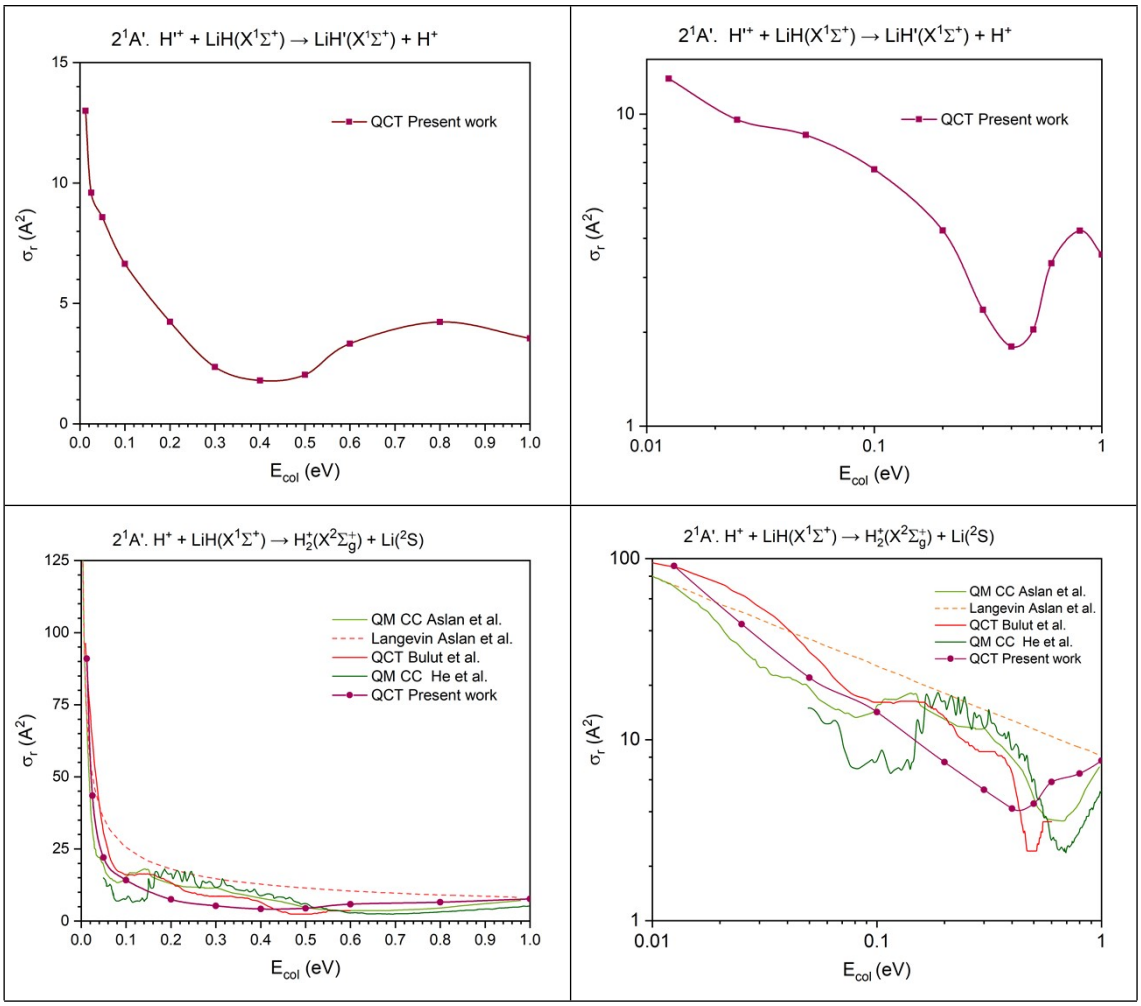

Figure S8. Cross sections for the reactions on the excited 2^1A' PES and comparison with other theoretical results (refs. 3, 10, 11): exchange (top) and H atom transfer (bottom). The double logarithmic representation is given on the right column.

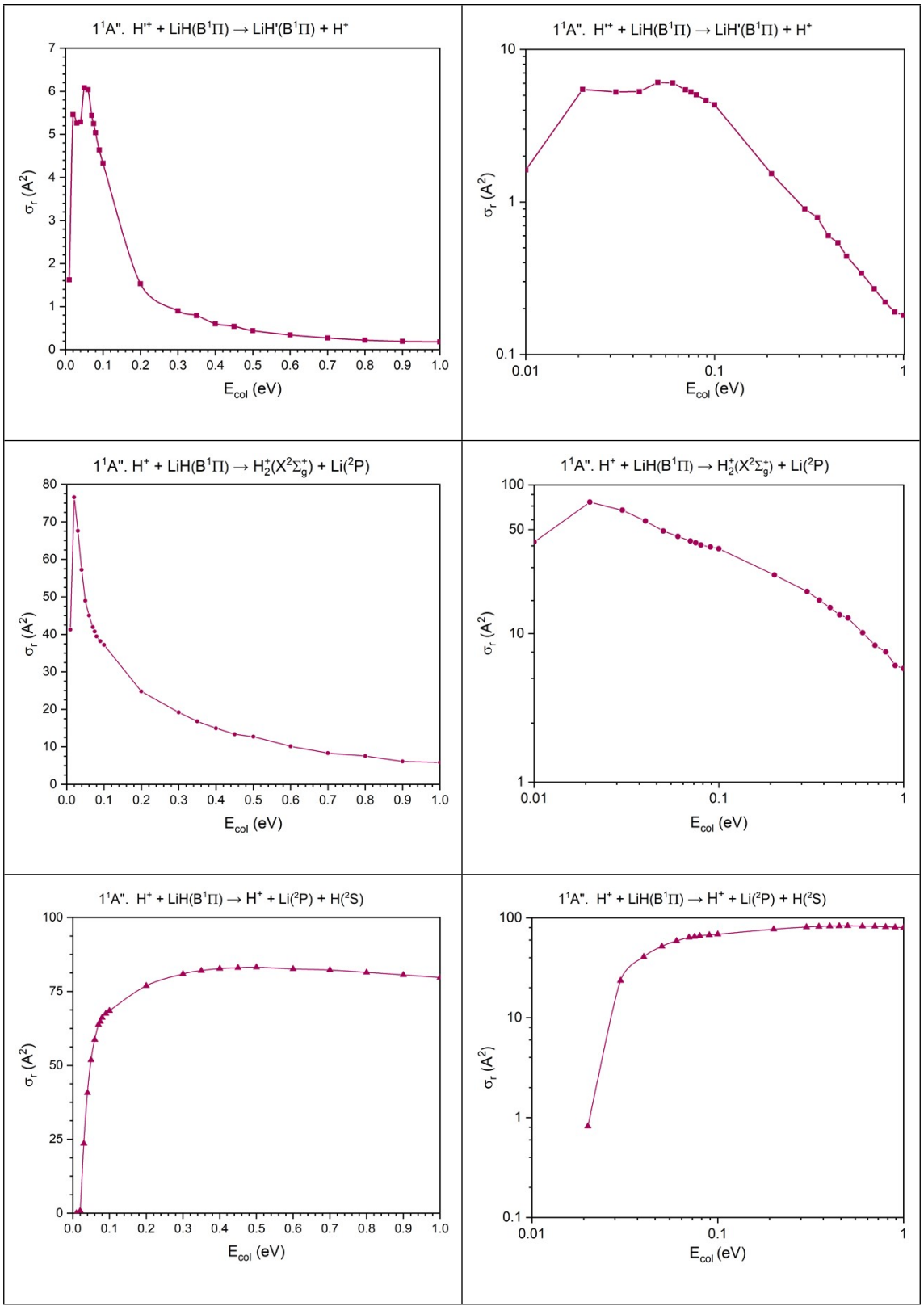

Figure S9. Cross sections for the reactions on the excited 1^1A" PES: exchange (top), H atom transfer (middle) and collision induced dissociation (bottom). The double logarithmic representation is given on the right column.

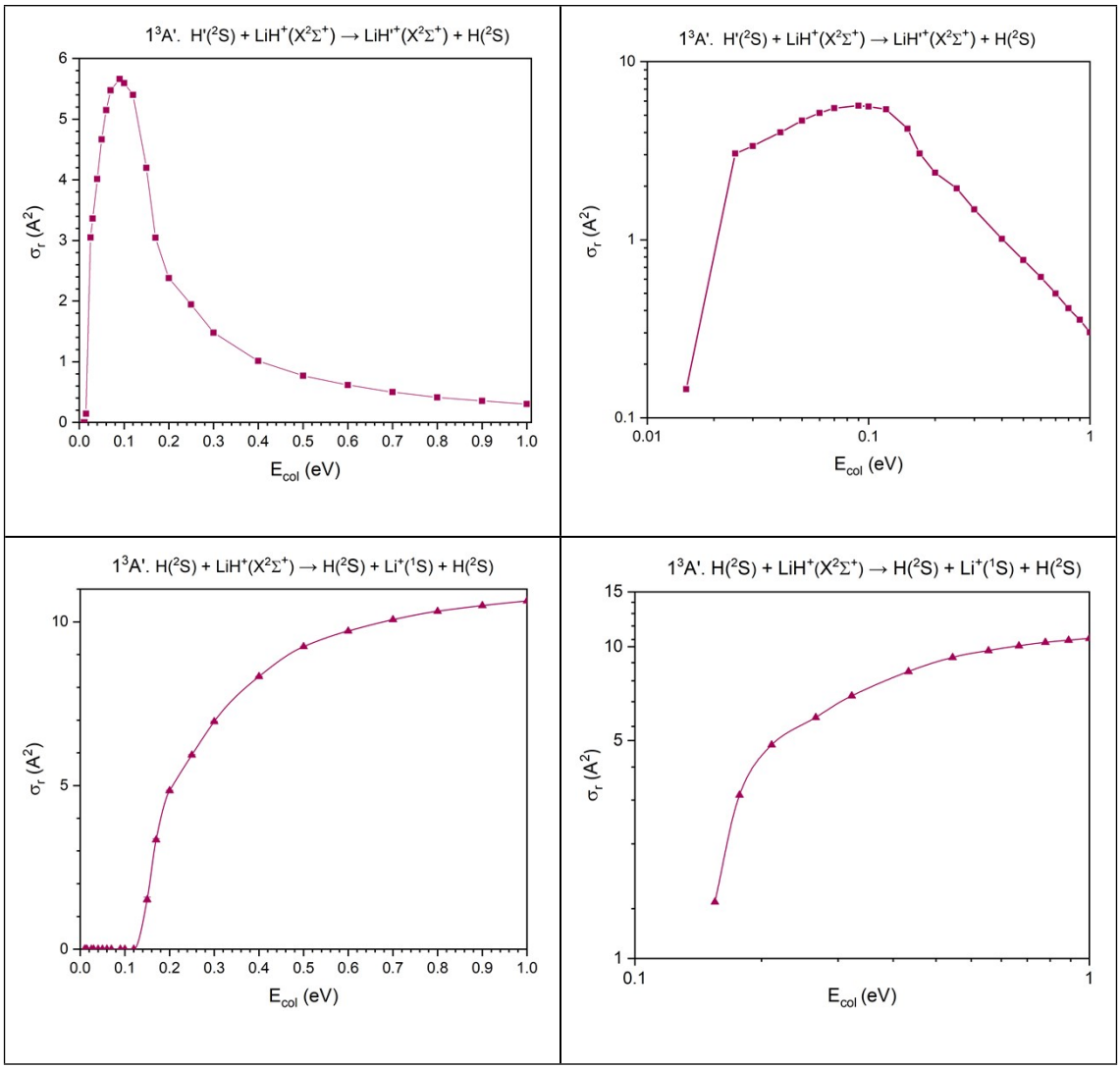

Figure S10. Cross sections for the reactions on the excited $1^3A'$ PES: exchange (top) and collision induced dissociation (bottom). The double logarithmic representation is given on the right column.

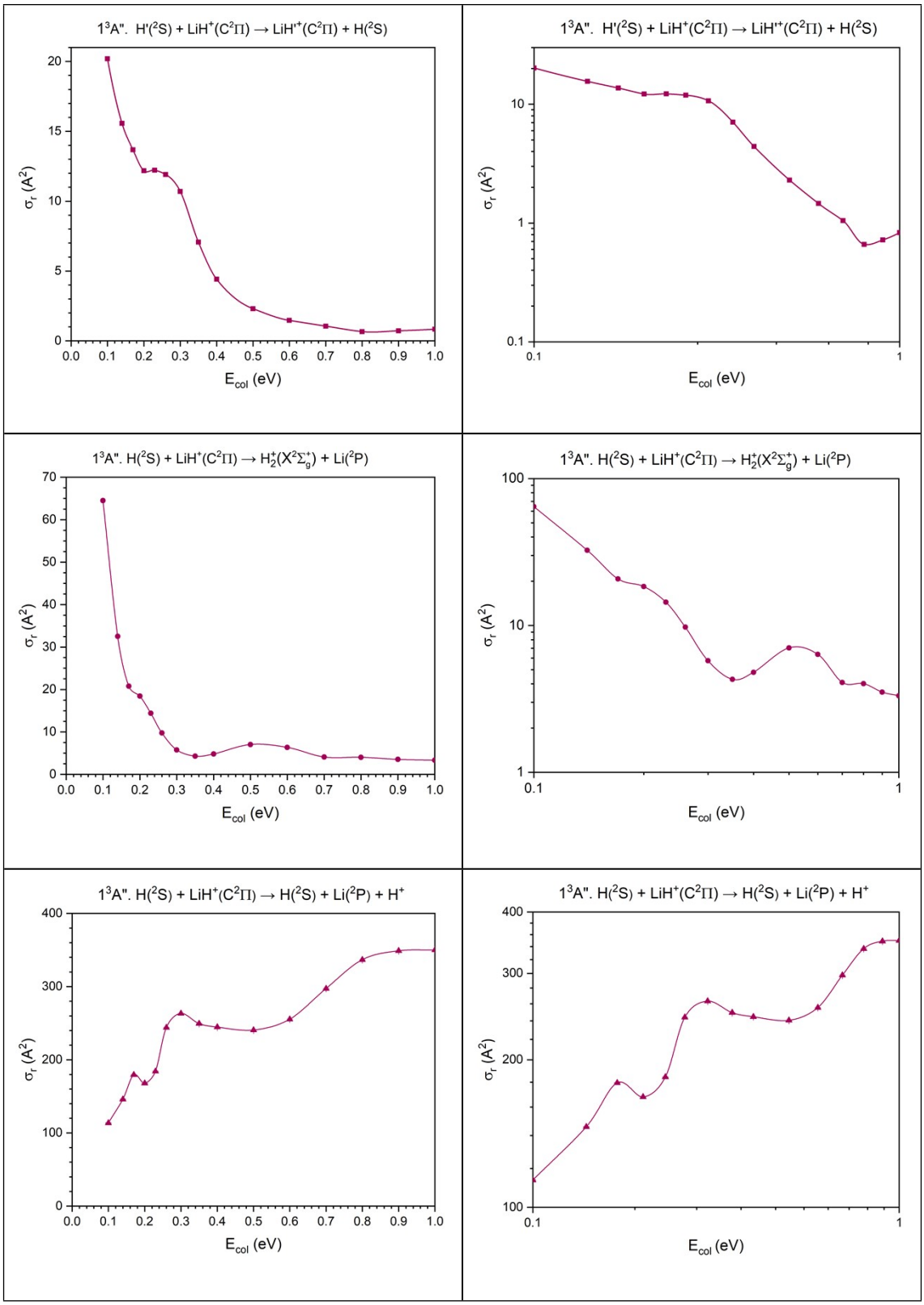

Figure S11. Cross sections for the reactions on the excited 1^3A" PES: exchange (top), H^+ transfer (middle) and collision induced dissociation (bottom). The double logarithmic representation is given on the right column.

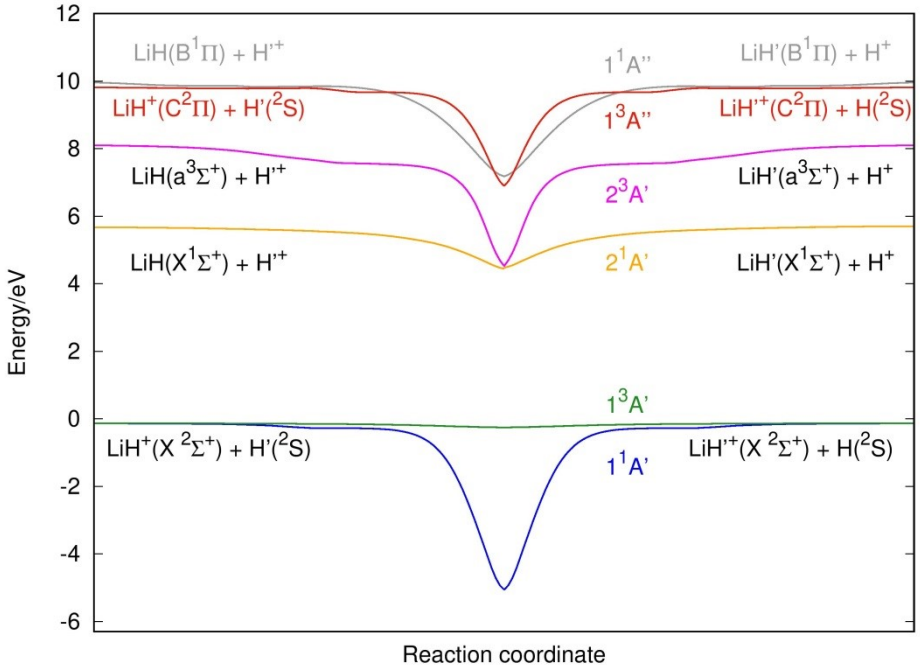

Figure S12. Schematic representation of the minimum energy paths for the exchange reaction channel. Energy is in eV with respect to the zero energy defined by the sum of the $Li^+(^1S)$ and twice $H(^2S)$ energies for all the six PESs considered. The reaction coordinate is given in arbitrary units.

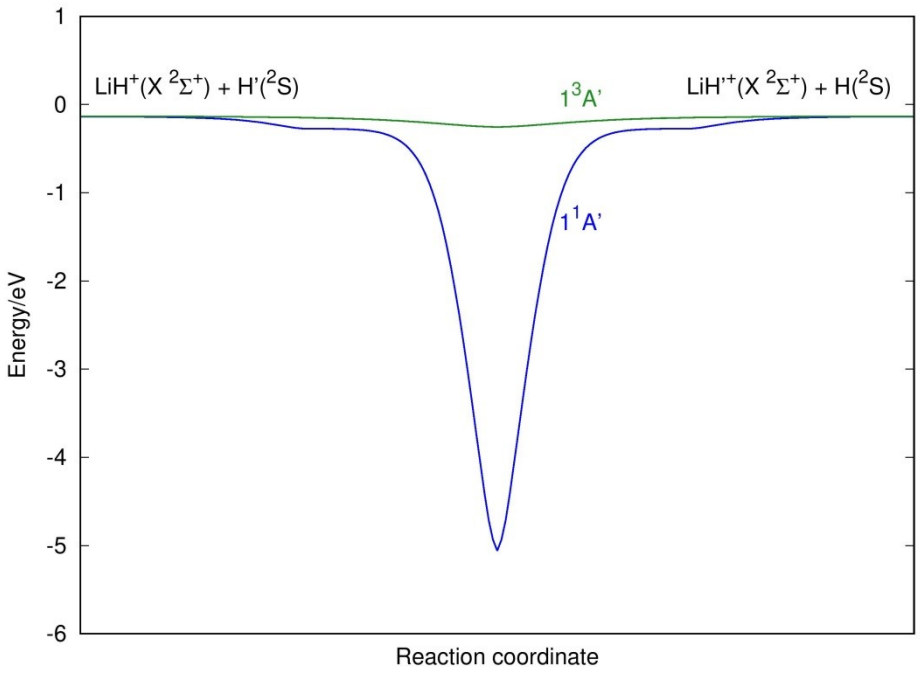

Figure S13. A more detailed view of the minimum energy paths for the exchange reaction channel of the 1^1A' and 1^3A' PESs. Energy is in eV with respect to zero energy defined by the sum of the $Li^+(^1S)$ and twice $H(^2S)$ energies. The reaction coordinate is given in arbitrary units.

References

[1] R. Martinazzo, G. F. Tantardini, E. Bodo and F. A. Gianturco, J. Chem. Phys. 2003, **119**, 11241.

[2] W. P. Kraemer and V. Ŝpirko, Chem. Phys. 2006, **330**, 190.

[3] X. He, S. Lv, T. Hayat and K. Han, J. Phys. Chem. A 2016, **120**, 2459.

[4] D. J. Searles and E. I. von Nagy-Felsobuki, Phys. Rev. A 1991, **43**, 3365.

[5] I. Pino, R. Martinazzo and G. F. Tantardini, Phys. Chem. Chem. Phys. 2008, **10**, 5545.

[6] T. Roy and S. Mahapatra, Chem. Phys. 2015, **448**, 34

[7] M. Dong, W. Li, D. He and M. Chen, RSC Adv. 2017, **7**, 7008.

[8] Y.-M. Li and Y. Lei, J. Theor. Comput. Chem. 2020, **19**, 2050002.

[9] J. Sahoo, A. M. S. Rawat and S. Mahapatra, J. Phys. Chem. A 2021, **125**, 3387.

[10] N. Bulut, J. F. Castillo, F. J. Aoiz and L. Bañares, Phys. Chem. Chem. Phys. 2008, **10**, 821.

[11] E. Aslan, N. Bulut, J. F. Castillo, L. Bañares, F. J. Aoiz and O. Roncero, Astrophys. J. 2012, **759**, 31.

Fortran code 1. Analytical expression of the ground 1^1A' PES**.**

```
subroutine fit3d(r12,r13,r23,e,der)
      implicit real*8 (a-h,o-z)
      dimension der(3)
      call diat12(r12,e12,d12)
      call diat12(r13,e13,d13)
      call diat23(r23,e23,d23)
      call triabb(r12,r13,r23,e123,der)
      e=e12+e13+e23+e123
      der(1)=d12+der(1)
      der(2)=d13+der(2)
      der(3)=d23+der(3)
      return
      end
*****************************************************************
********
      subroutine diat12(r,ener,der)
*****************************************************************
********
*     This subroutine computes the energies of a diatomic
potential
*     fitted to    54 points
*     rms =      0.00962617 kcal/mol
*     emax =      0.02792475 kcal/mol
*****************************************************************
********
      implicit real*8 (a-h,o-z)
      dimension cf(  7)
      data cf(  1)/0.1440890733503789D+01/
      data cf(  2)/-.5412322695150536D-01/
      data cf(  3)/-.4163802944217001D+00/
      data cf(  4)/0.8576852475109086D-01/
      data cf(  5)/0.3309760270606295D+00/
      data cf(  6)/-.7977077554345092D+00/
      data cf(  7)/0.2451531836339704D+00/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.6991300994658561D+00
      vex2=0.6119847074786407D+00
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2,  7
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
```

```
      end
*************************************************************
********
      subroutine diat23(r,ener,der)
*************************************************************
********
*     This subroutine computes the energies of a diatomic
potential
*     fitted to    59 points
*     rms =      0.03804557 kcal/mol
*     emax =      0.09823988 kcal/mol
*************************************************************
********
      implicit real*8 (a-h,o-z)
      dimension cf(  9)
      data cf(  1)/0.7363322426706777D+00/
      data cf(  2)/-.3720297141830095D+01/
      data cf(  3)/0.2561632528249042D+02/
      data cf(  4)/-.1889752559185310D+03/
      data cf(  5)/0.3963507253283522D+03/
      data cf(  6)/0.5652927233516857D+04/
      data cf(  7)/-.5310607688527115D+05/
      data cf(  8)/0.1842872473224173D+06/
      data cf(  9)/-.2390105816282381D+06/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.1612539122145108D+01
      vex2=0.8525862854625745D+00
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2,  9
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
*************************************************************
      subroutine triabb(r12,r13,r23,ener,der)
*************************************************************
*     This subroutine computes the energies of a 3D PES
*     for the ABB system class fitted to 44612 points
*     rms =      0.37268159 kcal/mol
*     emax =      3.51520088 kcal/mol
*************************************************************
      implicit real*8(a-h,o-z)
      dimension i1(   55),i2(   55),i3(   55),i4(   55),cf(
55)
      dimension f12(0: 6),f13(0: 6),f23(0: 6)
      dimension der(3)
```

```
data cf(  1)/-.1014547220695584D+01/
data i1(  1)/ 1/,i2(  1)/ 1/,i3(  1)/ 0/,i4(  1)/ 2/
data cf(  2)/-.2794317684912635D+01/
data i1(  2)/ 1/,i2(  2)/ 0/,i3(  2)/ 1/,i4(  2)/ 1/
data cf(  3)/-.1053266019598949D+03/
data i1(  3)/ 1/,i2(  3)/ 1/,i3(  3)/ 1/,i4(  3)/ 1/
data cf(  4)/0.2171803906723379D+02/
data i1(  4)/ 2/,i2(  4)/ 1/,i3(  4)/ 0/,i4(  4)/ 2/
data cf(  5)/0.4657322470946350D+02/
data i1(  5)/ 2/,i2(  5)/ 0/,i3(  5)/ 1/,i4(  5)/ 2/
data cf(  6)/0.2262336309104361D+02/
data i1(  6)/ 0/,i2(  6)/ 2/,i3(  6)/ 1/,i4(  6)/ 2/
data cf(  7)/0.7464121744663421D+03/
data i1(  7)/ 2/,i2(  7)/ 1/,i3(  7)/ 1/,i4(  7)/ 2/
data cf(  8)/-.1588418796740669D+03/
data i1(  8)/ 1/,i2(  8)/ 2/,i3(  8)/ 1/,i4(  8)/ 1/
data cf(  9)/-.9171218688960189D+02/
data i1(  9)/ 2/,i2(  9)/ 2/,i3(  9)/ 0/,i4(  9)/ 2/
data cf( 10)/-.5362573512252783D+03/
data i1( 10)/ 2/,i2( 10)/ 0/,i3( 10)/ 2/,i4( 10)/ 1/
data cf( 11)/-.2179909292013345D+03/
data i1( 11)/ 3/,i2( 11)/ 1/,i3( 11)/ 0/,i4( 11)/ 2/
data cf( 12)/-.2884080497409670D+03/
data i1( 12)/ 3/,i2( 12)/ 0/,i3( 12)/ 1/,i4( 12)/ 2/
data cf( 13)/0.3974211254489363D+02/
data i1( 13)/ 0/,i2( 13)/ 3/,i3( 13)/ 1/,i4( 13)/ 2/
data cf( 14)/-.1793690683794917D+04/
data i1( 14)/ 2/,i2( 14)/ 2/,i3( 14)/ 1/,i4( 14)/ 2/
data cf( 15)/-.3864892369796088D+04/
data i1( 15)/ 2/,i2( 15)/ 1/,i3( 15)/ 2/,i4( 15)/ 1/
data cf( 16)/-.1318993295477847D+04/
data i1( 16)/ 3/,i2( 16)/ 1/,i3( 16)/ 1/,i4( 16)/ 2/
data cf( 17)/0.2857943078833065D+04/
data i1( 17)/ 1/,i2( 17)/ 3/,i3( 17)/ 1/,i4( 17)/ 1/
data cf( 18)/0.2586627239815855D+03/
data i1( 18)/ 3/,i2( 18)/ 2/,i3( 18)/ 0/,i4( 18)/ 2/
data cf( 19)/0.2358772235582289D+04/
data i1( 19)/ 3/,i2( 19)/ 0/,i3( 19)/ 2/,i4( 19)/ 2/
data cf( 20)/0.7009631599549559D+03/
data i1( 20)/ 0/,i2( 20)/ 3/,i3( 20)/ 2/,i4( 20)/ 2/
data cf( 21)/0.6290723089078425D+03/
data i1( 21)/ 4/,i2( 21)/ 1/,i3( 21)/ 0/,i4( 21)/ 2/
data cf( 22)/0.8014934692105323D+03/
data i1( 22)/ 4/,i2( 22)/ 0/,i3( 22)/ 1/,i4( 22)/ 2/
data cf( 23)/-.1618135786346971D+04/
data i1( 23)/ 0/,i2( 23)/ 4/,i3( 23)/ 1/,i4( 23)/ 2/
data cf( 24)/0.6134395961989920D+04/
data i1( 24)/ 2/,i2( 24)/ 2/,i3( 24)/ 2/,i4( 24)/ 1/
data cf( 25)/0.1936241598644449D+04/
data i1( 25)/ 3/,i2( 25)/ 2/,i3( 25)/ 1/,i4( 25)/ 2/
data cf( 26)/0.6337895967561134D+04/
data i1( 26)/ 3/,i2( 26)/ 1/,i3( 26)/ 2/,i4( 26)/ 2/
data cf( 27)/0.2018186679116811D+04/
data i1( 27)/ 1/,i2( 27)/ 3/,i3( 27)/ 2/,i4( 27)/ 2/
data cf( 28)/0.2714246624469634D+04/
data i1( 28)/ 3/,i2( 28)/ 3/,i3( 28)/ 0/,i4( 28)/ 2/
```

```
data cf( 29)/-.6087465622502077D+04/
data i1( 29)/ 3/,i2( 29)/ 0/,i3( 29)/ 3/,i4( 29)/ 1/
data cf( 30)/0.3722476143730888D+03/
data i1( 30)/ 4/,i2( 30)/ 1/,i3( 30)/ 1/,i4( 30)/ 2/
data cf( 31)/-.8625308023933826D+04/
data i1( 31)/ 1/,i2( 31)/ 4/,i3( 31)/ 1/,i4( 31)/ 1/
data cf( 32)/-.2413305858257138D+04/
data i1( 32)/ 4/,i2( 32)/ 2/,i3( 32)/ 0/,i4( 32)/ 2/
data cf( 33)/-.4527548814071714D+04/
data i1( 33)/ 4/,i2( 33)/ 0/,i3( 33)/ 2/,i4( 33)/ 2/
data cf( 34)/-.4361164524816215D+04/
data i1( 34)/ 0/,i2( 34)/ 4/,i3( 34)/ 2/,i4( 34)/ 2/
data cf( 35)/-.2173173418381897D+02/
data i1( 35)/ 5/,i2( 35)/ 1/,i3( 35)/ 0/,i4( 35)/ 2/
data cf( 36)/-.9124696176573320D+03/
data i1( 36)/ 5/,i2( 36)/ 0/,i3( 36)/ 1/,i4( 36)/ 2/
data cf( 37)/0.7738466184538001D+04/
data i1( 37)/ 0/,i2( 37)/ 5/,i3( 37)/ 1/,i4( 37)/ 2/
data cf( 38)/-.6715658741369955D+03/
data i1( 38)/ 3/,i2( 38)/ 2/,i3( 38)/ 2/,i4( 38)/ 2/
data cf( 39)/-.2024021230760642D+05/
data i1( 39)/ 2/,i2( 39)/ 3/,i3( 39)/ 2/,i4( 39)/ 1/
data cf( 40)/0.2284494631121076D+05/
data i1( 40)/ 3/,i2( 40)/ 3/,i3( 40)/ 1/,i4( 40)/ 2/
data cf( 41)/-.2451588941089384D+05/
data i1( 41)/ 3/,i2( 41)/ 1/,i3( 41)/ 3/,i4( 41)/ 1/
data cf( 42)/-.1796836787388634D+05/
data i1( 42)/ 4/,i2( 42)/ 2/,i3( 42)/ 1/,i4( 42)/ 2/
data cf( 43)/0.1093420805120413D+05/
data i1( 43)/ 4/,i2( 43)/ 1/,i3( 43)/ 2/,i4( 43)/ 2/
data cf( 44)/-.7867188429861572D+04/
data i1( 44)/ 1/,i2( 44)/ 4/,i3( 44)/ 2/,i4( 44)/ 2/
data cf( 45)/-.2093164311965843D+04/
data i1( 45)/ 4/,i2( 45)/ 3/,i3( 45)/ 0/,i4( 45)/ 2/
data cf( 46)/0.5348596796357068D+04/
data i1( 46)/ 4/,i2( 46)/ 0/,i3( 46)/ 3/,i4( 46)/ 2/
data cf( 47)/-.6318793851052782D+04/
data i1( 47)/ 0/,i2( 47)/ 4/,i3( 47)/ 3/,i4( 47)/ 2/
data cf( 48)/0.2434449562014364D+03/
data i1( 48)/ 5/,i2( 48)/ 1/,i3( 48)/ 1/,i4( 48)/ 2/
data cf( 49)/0.1107542831535004D+05/
data i1( 49)/ 1/,i2( 49)/ 5/,i3( 49)/ 1/,i4( 49)/ 1/
data cf( 50)/0.5859854181305584D+04/
data i1( 50)/ 5/,i2( 50)/ 2/,i3( 50)/ 0/,i4( 50)/ 2/
data cf( 51)/0.2649597606540259D+04/
data i1( 51)/ 5/,i2( 51)/ 0/,i3( 51)/ 2/,i4( 51)/ 2/
data cf( 52)/0.8608878093156504D+04/
data i1( 52)/ 0/,i2( 52)/ 5/,i3( 52)/ 2/,i4( 52)/ 2/
data cf( 53)/-.1658147436455788D+04/
data i1( 53)/ 6/,i2( 53)/ 1/,i3( 53)/ 0/,i4( 53)/ 2/
data cf( 54)/0.2962363762398714D+03/
data i1( 54)/ 6/,i2( 54)/ 0/,i3( 54)/ 1/,i4( 54)/ 2/
data cf( 55)/-.1150180303193682D+05/
data i1( 55)/ 0/,i2( 55)/ 6/,i3( 55)/ 1/,i4( 55)/ 2/
vex1=0.1027980066618575D+01
vex2=0.1372494569355204D+01
```

```
      f12(0)=1.d0
      f13(0)=1.d0
      f23(0)=1.d0
      bux12=r12*dexp(-vex1*r12)
      bux13=r13*dexp(-vex1*r13)
      bux23=r23*dexp(-vex2*r23)
      do 1 i=1, 6
         f12(i)=f12(i-1)*bux12
         f13(i)=f13(i-1)*bux13
         f23(i)=f23(i-1)*bux23
1     continue
      ener = 0.d0
      der12 = 0.d0
      der13 = 0.d0
      der23 = 0.d0
      do 2 l=1, 55
         if (i4(l).eq.1) then
            aux=f12(i1(l))*f13(i3(l))*f23(i2(l))
            dux12=i1(l)*f12(i1(l)-1)*f13(i3(l))*f23(i2(l))
            dux13=i3(l)*f12(i1(l))*f13(i3(l)-1)*f23(i2(l))
            dux23=i2(l)*f12(i1(l))*f13(i3(l))*f23(i2(l)-1)
         else
            aux1=f12(i1(l))*f13(i3(l))
            aux2=f12(i3(l))*f13(i1(l))
            aux=(aux1+aux2)*f23(i2(l))
            dux23=(aux1+aux2)*i2(l)*f23(i2(l)-1)
            dux1=i1(l)*f12(i1(l)-1)*f13(i3(l))
            dux2=i3(l)*f12(i3(l)-1)*f13(i1(l))
            dux12=(dux1+dux2)*f23(i2(l))
            dux1=i3(l)*f12(i1(l))*f13(i3(l)-1)
            dux2=i1(l)*f12(i3(l))*f13(i1(l)-1)
            dux13=(dux1+dux2)*f23(i2(l))
         endif
         ener=ener+cf(l)*aux
         der12=der12+cf(l)*dux12
         der13=der13+cf(l)*dux13
         der23=der23+cf(l)*dux23
    2 continue
      der(1)=der12*(1.d0-vex1*r12)*dexp(-vex1*r12)
      der(2)=der13*(1.d0-vex1*r13)*dexp(-vex1*r13)
      der(3)=der23*(1.d0-vex2*r23)*dexp(-vex2*r23)
      return
      end
```

Fortran code 2. Analytical expression of the excited $2^1A'$ PES.

```
subroutine fit3d(r12,r13,r23,e,der)
      implicit real*8 (a-h,o-z)
      dimension der(3)
      call diat12(r12,e12,d12)
      call diat12(r13,e13,d13)
      call diat23(r23,e23,d23)
      call triabb(r12,r13,r23,e123,der)
      e=e12+e13+e23+e123
      der(1)=d12+der(1)
      der(2)=d13+der(2)
      der(3)=d23+der(3)
      return
      end
*************************************************************
********
      subroutine diat12(r,ener,der)
*************************************************************
********
*     This subroutine computes the energies of a diatomic
potential
*     fitted to    53 points
*     rms =      0.72758578 kcal/mol
*     emax =      1.39552895 kcal/mol
*************************************************************
********
      implicit real*8 (a-h,o-z)
      dimension cf( 13)
      data cf(  1)/0.252306455875266 9D+01/
      data cf(  2)/-.6402706666014535D+01/
      data cf(  3)/0.3587056876797626D+03/
      data cf(  4)/-.1434349390256489D+05/
      data cf(  5)/0.3599481614963323D+06/
      data cf(  6)/-.5782698778156763D+07/
      data cf(  7)/0.6148002385585026D+08/
      data cf(  8)/-.4414318497698402D+09/
      data cf(  9)/0.2149750038999408D+10/
      data cf( 10)/-.6994564632141600D+10/
      data cf( 11)/0.1455226646768256D+11/
      data cf( 12)/-.1750087151748045D+11/
      data cf( 13)/0.9252426397575498D+10/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.1168260087394666D+01
      vex2=0.1748386264688349D+01
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2, 13
```

```
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
*************************************************************
********
      subroutine diat23(r,ener,der)
*************************************************************
********
*     This subroutine computes the energies of a diatomic
potential
*     fitted to    69 points
*     rms =      0.00678127 kcal/mol
*     emax =      0.01986148 kcal/mol
*************************************************************
********
      implicit real*8 (a-h,o-z)
      dimension cf( 11)
      data cf(  1)/0.1023072995011111D+01/
      data cf(  2)/-.2416709176408609D+00/
      data cf(  3)/-.3900969316074534D+01/
      data cf(  4)/0.6709516064703307D+02/
      data cf(  5)/-.6913716386068441D+03/
      data cf(  6)/0.4585460355375840D+04/
      data cf(  7)/-.1983923520023503D+05/
      data cf(  8)/0.5554812388226776D+05/
      data cf(  9)/-.9691888604977434D+05/
      data cf( 10)/0.9572866154008034D+05/
      data cf( 11)/-.4086601888467368D+05/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.8492838719300601D+00
      vex2=0.1706755264515816D+01
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2, 11
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
*************************************************************
      subroutine triabb(r12,r13,r23,ener,der)
*************************************************************
*     This subroutine computes the energies of a 3D PES
```

```
*     for the ABB system class fitted to  49153 points
*     rms =      0.78978166 kcal/mol
*     emax =     11.25565406 kcal/mol
************************************************************
      implicit real*8(a-h,o-z)
      dimension i1(  106),i2(  106),i3(  106),i4(  106),cf(
106)
      dimension f12(0: 8),f13(0: 8),f23(0: 8)
      dimension der(3)
      data cf(  1)/0.5722918820543362D+01/
      data i1(  1)/ 1/,i2(  1)/ 1/,i3(  1)/ 0/,i4(  1)/ 2/
      data cf(  2)/-.5017983707936421D+02/
      data i1(  2)/ 1/,i2(  2)/ 0/,i3(  2)/ 1/,i4(  2)/ 1/
      data cf(  3)/0.2091355443252302D+03/
      data i1(  3)/ 1/,i2(  3)/ 1/,i3(  3)/ 1/,i4(  3)/ 1/
      data cf(  4)/-.2456003873515218D+02/
      data i1(  4)/ 2/,i2(  4)/ 1/,i3(  4)/ 0/,i4(  4)/ 2/
      data cf(  5)/0.1173692936103162D+03/
      data i1(  5)/ 2/,i2(  5)/ 0/,i3(  5)/ 1/,i4(  5)/ 2/
      data cf(  6)/-.2747991963821448D+02/
      data i1(  6)/ 0/,i2(  6)/ 2/,i3(  6)/ 1/,i4(  6)/ 2/
      data cf(  7)/-.5046719947608118D+03/
      data i1(  7)/ 2/,i2(  7)/ 1/,i3(  7)/ 1/,i4(  7)/ 2/
      data cf(  8)/-.3232564522392671D+03/
      data i1(  8)/ 1/,i2(  8)/ 2/,i3(  8)/ 1/,i4(  8)/ 1/
      data cf(  9)/0.8420240314681237D+02/
      data i1(  9)/ 2/,i2(  9)/ 2/,i3(  9)/ 0/,i4(  9)/ 2/
      data cf( 10)/-.1456568046999063D+03/
      data i1( 10)/ 2/,i2( 10)/ 0/,i3( 10)/ 2/,i4( 10)/ 1/
      data cf( 11)/0.8111830392050980D+02/
      data i1( 11)/ 3/,i2( 11)/ 1/,i3( 11)/ 0/,i4( 11)/ 2/
      data cf( 12)/0.1814387458339129D+02/
      data i1( 12)/ 3/,i2( 12)/ 0/,i3( 12)/ 1/,i4( 12)/ 2/
      data cf( 13)/0.5151532224914611D+02/
      data i1( 13)/ 0/,i2( 13)/ 3/,i3( 13)/ 1/,i4( 13)/ 2/
      data cf( 14)/0.5636038984701948D+03/
      data i1( 14)/ 2/,i2( 14)/ 2/,i3( 14)/ 1/,i4( 14)/ 2/
      data cf( 15)/0.1133929076992068D+04/
      data i1( 15)/ 2/,i2( 15)/ 1/,i3( 15)/ 2/,i4( 15)/ 1/
      data cf( 16)/0.5012583153185484D+03/
      data i1( 16)/ 3/,i2( 16)/ 1/,i3( 16)/ 1/,i4( 16)/ 2/
      data cf( 17)/0.2880469687959027D+03/
      data i1( 17)/ 1/,i2( 17)/ 3/,i3( 17)/ 1/,i4( 17)/ 1/
      data cf( 18)/-.1145283430311228D+03/
      data i1( 18)/ 3/,i2( 18)/ 2/,i3( 18)/ 0/,i4( 18)/ 2/
      data cf( 19)/-.4113039112545837D+03/
      data i1( 19)/ 3/,i2( 19)/ 0/,i3( 19)/ 2/,i4( 19)/ 2/
      data cf( 20)/-.1329220998911589D+03/
      data i1( 20)/ 0/,i2( 20)/ 3/,i3( 20)/ 2/,i4( 20)/ 2/
      data cf( 21)/-.2204329015530753D+03/
      data i1( 21)/ 4/,i2( 21)/ 1/,i3( 21)/ 0/,i4( 21)/ 2/
      data cf( 22)/-.4864662003404308D+03/
      data i1( 22)/ 4/,i2( 22)/ 0/,i3( 22)/ 1/,i4( 22)/ 2/
      data cf( 23)/-.4986782955975617D+02/
      data i1( 23)/ 0/,i2( 23)/ 4/,i3( 23)/ 1/,i4( 23)/ 2/
      data cf( 24)/-.8272430897170308D+03/
```

```
data i1( 24)/ 2/,i2( 24)/ 2/,i3( 24)/ 2/,i4( 24)/ 1/
data cf( 25)/-.6796345823158415D+03/
data i1( 25)/ 3/,i2( 25)/ 2/,i3( 25)/ 1/,i4( 25)/ 2/
data cf( 26)/-.1035454885634819D+04/
data i1( 26)/ 3/,i2( 26)/ 1/,i3( 26)/ 2/,i4( 26)/ 2/
data cf( 27)/-.3016998659685465D+03/
data i1( 27)/ 1/,i2( 27)/ 3/,i3( 27)/ 2/,i4( 27)/ 2/
data cf( 28)/0.1098369733408290D+03/
data i1( 28)/ 3/,i2( 28)/ 3/,i3( 28)/ 0/,i4( 28)/ 2/
data cf( 29)/0.4098740303614485D+04/
data i1( 29)/ 3/,i2( 29)/ 0/,i3( 29)/ 3/,i4( 29)/ 1/
data cf( 30)/0.3396934472768041D+03/
data i1( 30)/ 4/,i2( 30)/ 1/,i3( 30)/ 1/,i4( 30)/ 2/
data cf( 31)/-.1645372199557781D+03/
data i1( 31)/ 1/,i2( 31)/ 4/,i3( 31)/ 1/,i4( 31)/ 1/
data cf( 32)/0.1549509667559394D+03/
data i1( 32)/ 4/,i2( 32)/ 2/,i3( 32)/ 0/,i4( 32)/ 2/
data cf( 33)/-.7226408049205251D+03/
data i1( 33)/ 4/,i2( 33)/ 0/,i3( 33)/ 2/,i4( 33)/ 2/
data cf( 34)/0.1108206488564541D+03/
data i1( 34)/ 0/,i2( 34)/ 4/,i3( 34)/ 2/,i4( 34)/ 2/
data cf( 35)/0.3112742652717123D+03/
data i1( 35)/ 5/,i2( 35)/ 1/,i3( 35)/ 0/,i4( 35)/ 2/
data cf( 36)/0.1289900741008550D+04/
data i1( 36)/ 5/,i2( 36)/ 0/,i3( 36)/ 1/,i4( 36)/ 2/
data cf( 37)/0.2659969707297347D+02/
data i1( 37)/ 0/,i2( 37)/ 5/,i3( 37)/ 1/,i4( 37)/ 2/
data cf( 38)/0.8116981400954038D+03/
data i1( 38)/ 3/,i2( 38)/ 2/,i3( 38)/ 2/,i4( 38)/ 2/
data cf( 39)/0.1182516249420435D+03/
data i1( 39)/ 2/,i2( 39)/ 3/,i3( 39)/ 2/,i4( 39)/ 1/
data cf( 40)/0.3478765640446728D+03/
data i1( 40)/ 3/,i2( 40)/ 3/,i3( 40)/ 1/,i4( 40)/ 2/
data cf( 41)/-.1778638593260127D+04/
data i1( 41)/ 3/,i2( 41)/ 1/,i3( 41)/ 3/,i4( 41)/ 1/
data cf( 42)/0.1866360615522683D+03/
data i1( 42)/ 4/,i2( 42)/ 2/,i3( 42)/ 1/,i4( 42)/ 2/
data cf( 43)/0.1758405344135222D+04/
data i1( 43)/ 4/,i2( 43)/ 1/,i3( 43)/ 2/,i4( 43)/ 2/
data cf( 44)/0.9699594346118272D+02/
data i1( 44)/ 1/,i2( 44)/ 4/,i3( 44)/ 2/,i4( 44)/ 2/
data cf( 45)/-.3390284281888744D+02/
data i1( 45)/ 4/,i2( 45)/ 3/,i3( 45)/ 0/,i4( 45)/ 2/
data cf( 46)/-.3431010583909843D+04/
data i1( 46)/ 4/,i2( 46)/ 0/,i3( 46)/ 3/,i4( 46)/ 2/
data cf( 47)/-.6946364932241389D+02/
data i1( 47)/ 0/,i2( 47)/ 4/,i3( 47)/ 3/,i4( 47)/ 2/
data cf( 48)/-.1505611667814908D+04/
data i1( 48)/ 5/,i2( 48)/ 1/,i3( 48)/ 1/,i4( 48)/ 2/
data cf( 49)/0.5826227659727804D+02/
data i1( 49)/ 1/,i2( 49)/ 5/,i3( 49)/ 1/,i4( 49)/ 1/
data cf( 50)/-.2170025736048315D+03/
data i1( 50)/ 5/,i2( 50)/ 2/,i3( 50)/ 0/,i4( 50)/ 2/
data cf( 51)/0.1775470915359418D+04/
data i1( 51)/ 5/,i2( 51)/ 0/,i3( 51)/ 2/,i4( 51)/ 2/
data cf( 52)/-.5033640032877160D+02/
```

```
data i1( 52)/ 0/,i2( 52)/ 5/,i3( 52)/ 2/,i4( 52)/ 2/
data cf( 53)/-.1265260545206771D+03/
data i1( 53)/ 6/,i2( 53)/ 1/,i3( 53)/ 0/,i4( 53)/ 2/
data cf( 54)/-.1550916995938528D+04/
data i1( 54)/ 6/,i2( 54)/ 0/,i3( 54)/ 1/,i4( 54)/ 2/
data cf( 55)/-.7508980784707570D+01/
data i1( 55)/ 0/,i2( 55)/ 6/,i3( 55)/ 1/,i4( 55)/ 2/
data cf( 56)/-.2045835052483064D+03/
data i1( 56)/ 3/,i2( 56)/ 3/,i3( 56)/ 2/,i4( 56)/ 2/
data cf( 57)/0.4742222069308286D+03/
data i1( 57)/ 3/,i2( 57)/ 2/,i3( 57)/ 3/,i4( 57)/ 1/
data cf( 58)/-.7195645248486708D+03/
data i1( 58)/ 4/,i2( 58)/ 2/,i3( 58)/ 2/,i4( 58)/ 2/
data cf( 59)/0.5439673991269598D+02/
data i1( 59)/ 2/,i2( 59)/ 4/,i3( 59)/ 2/,i4( 59)/ 1/
data cf( 60)/-.1074683563897579D+03/
data i1( 60)/ 4/,i2( 60)/ 3/,i3( 60)/ 1/,i4( 60)/ 2/
data cf( 61)/0.5879884160147818D+03/
data i1( 61)/ 4/,i2( 61)/ 1/,i3( 61)/ 3/,i4( 61)/ 2/
data cf( 62)/-.8636293702511852D+02/
data i1( 62)/ 1/,i2( 62)/ 4/,i3( 62)/ 3/,i4( 62)/ 2/
data cf( 63)/0.1409899845617880D+02/
data i1( 63)/ 4/,i2( 63)/ 4/,i3( 63)/ 0/,i4( 63)/ 2/
data cf( 64)/0.2323404164990046D+04/
data i1( 64)/ 4/,i2( 64)/ 0/,i3( 64)/ 4/,i4( 64)/ 1/
data cf( 65)/0.2153556996033233D+03/
data i1( 65)/ 5/,i2( 65)/ 2/,i3( 65)/ 1/,i4( 65)/ 2/
data cf( 66)/-.1001169410632592D+04/
data i1( 66)/ 5/,i2( 66)/ 1/,i3( 66)/ 2/,i4( 66)/ 2/
data cf( 67)/-.2051437660506561D+02/
data i1( 67)/ 1/,i2( 67)/ 5/,i3( 67)/ 2/,i4( 67)/ 2/
data cf( 68)/0.6109290077421992D+01/
data i1( 68)/ 5/,i2( 68)/ 3/,i3( 68)/ 0/,i4( 68)/ 2/
data cf( 69)/0.1504433697360542D+04/
data i1( 69)/ 5/,i2( 69)/ 0/,i3( 69)/ 3/,i4( 69)/ 2/
data cf( 70)/0.2104744467630383D+02/
data i1( 70)/ 0/,i2( 70)/ 5/,i3( 70)/ 3/,i4( 70)/ 2/
data cf( 71)/0.1423317775012512D+04/
data i1( 71)/ 6/,i2( 71)/ 1/,i3( 71)/ 1/,i4( 71)/ 2/
data cf( 72)/-.1128423842046185D+02/
data i1( 72)/ 1/,i2( 72)/ 6/,i3( 72)/ 1/,i4( 72)/ 1/
data cf( 73)/0.1694866673209051D+03/
data i1( 73)/ 6/,i2( 73)/ 2/,i3( 73)/ 0/,i4( 73)/ 2/
data cf( 74)/-.1705604967883224D+04/
data i1( 74)/ 6/,i2( 74)/ 0/,i3( 74)/ 2/,i4( 74)/ 2/
data cf( 75)/0.1192250267955902D+02/
data i1( 75)/ 0/,i2( 75)/ 6/,i3( 75)/ 2/,i4( 75)/ 2/
data cf( 76)/-.1106344925735777D+03/
data i1( 76)/ 7/,i2( 76)/ 1/,i3( 76)/ 0/,i4( 76)/ 2/
data cf( 77)/0.9539922178007842D+03/
data i1( 77)/ 7/,i2( 77)/ 0/,i3( 77)/ 1/,i4( 77)/ 2/
data cf( 78)/0.9156443170884776D+00/
data i1( 78)/ 0/,i2( 78)/ 7/,i3( 78)/ 1/,i4( 78)/ 2/
data cf( 79)/0.2202761048325887D+03/
data i1( 79)/ 3/,i2( 79)/ 3/,i3( 79)/ 3/,i4( 79)/ 1/
data cf( 80)/0.2166388132272332D+01/
```

```
data i1( 80)/ 4/,i2( 80)/ 3/,i3( 80)/ 2/,i4( 80)/ 2/
data cf( 81)/-.5218389250369101D+03/
data i1( 81)/ 4/,i2( 81)/ 2/,i3( 81)/ 3/,i4( 81)/ 2/
data cf( 82)/-.1008775347661615D+02/
data i1( 82)/ 2/,i2( 82)/ 4/,i3( 82)/ 3/,i4( 82)/ 2/
data cf( 83)/0.4479320280894132D+02/
data i1( 83)/ 4/,i2( 83)/ 4/,i3( 83)/ 1/,i4( 83)/ 2/
data cf( 84)/0.1177645997270097D+04/
data i1( 84)/ 4/,i2( 84)/ 1/,i3( 84)/ 4/,i4( 84)/ 1/
data cf( 85)/0.6685823938612132D+03/
data i1( 85)/ 5/,i2( 85)/ 2/,i3( 85)/ 2/,i4( 85)/ 2/
data cf( 86)/-.6572547883797575D+01/
data i1( 86)/ 2/,i2( 86)/ 5/,i3( 86)/ 2/,i4( 86)/ 1/
data cf( 87)/-.7079540500968668D+02/
data i1( 87)/ 5/,i2( 87)/ 3/,i3( 87)/ 1/,i4( 87)/ 2/
data cf( 88)/-.2516809643969990D+03/
data i1( 88)/ 5/,i2( 88)/ 1/,i3( 88)/ 3/,i4( 88)/ 2/
data cf( 89)/0.5271049526638937D+01/
data i1( 89)/ 1/,i2( 89)/ 5/,i3( 89)/ 3/,i4( 89)/ 2/
data cf( 90)/0.9520258088105281D+01/
data i1( 90)/ 5/,i2( 90)/ 4/,i3( 90)/ 0/,i4( 90)/ 2/
data cf( 91)/-.1121684082143045D+04/
data i1( 91)/ 5/,i2( 91)/ 0/,i3( 91)/ 4/,i4( 91)/ 2/
data cf( 92)/-.5425794825174498D+01/
data i1( 92)/ 0/,i2( 92)/ 5/,i3( 92)/ 4/,i4( 92)/ 2/
data cf( 93)/-.1287743158140319D+03/
data i1( 93)/ 6/,i2( 93)/ 2/,i3( 93)/ 1/,i4( 93)/ 2/
data cf( 94)/-.4383079802832761D+03/
data i1( 94)/ 6/,i2( 94)/ 1/,i3( 94)/ 2/,i4( 94)/ 2/
data cf( 95)/0.2117848136778486D+01/
data i1( 95)/ 1/,i2( 95)/ 6/,i3( 95)/ 2/,i4( 95)/ 2/
data cf( 96)/-.1835339764486768D+02/
data i1( 96)/ 6/,i2( 96)/ 3/,i3( 96)/ 0/,i4( 96)/ 2/
data cf( 97)/0.1601318568826274D+03/
data i1( 97)/ 6/,i2( 97)/ 0/,i3( 97)/ 3/,i4( 97)/ 2/
data cf( 98)/-.1965448963161335D+01/
data i1( 98)/ 0/,i2( 98)/ 6/,i3( 98)/ 3/,i4( 98)/ 2/
data cf( 99)/-.2955286609087560D+03/
data i1( 99)/ 7/,i2( 99)/ 1/,i3( 99)/ 1/,i4( 99)/ 2/
data cf(100)/0.9064482961225421D+00/
data i1(100)/ 1/,i2(100)/ 7/,i3(100)/ 1/,i4(100)/ 1/
data cf(101)/-.3723896205232873D+02/
data i1(101)/ 7/,i2(101)/ 2/,i3(101)/ 0/,i4(101)/ 2/
data cf(102)/0.9069253493005318D+03/
data i1(102)/ 7/,i2(102)/ 0/,i3(102)/ 2/,i4(102)/ 2/
data cf(103)/-.1187233028861015D+01/
data i1(103)/ 0/,i2(103)/ 7/,i3(103)/ 2/,i4(103)/ 2/
data cf(104)/0.8226337275619237D+02/
data i1(104)/ 8/,i2(104)/ 1/,i3(104)/ 0/,i4(104)/ 2/
data cf(105)/-.3977161750734615D+03/
data i1(105)/ 8/,i2(105)/ 0/,i3(105)/ 1/,i4(105)/ 2/
data cf(106)/-.1435949801245581D-01/
data i1(106)/ 0/,i2(106)/ 8/,i3(106)/ 1/,i4(106)/ 2/
vex1=0.4996307990298401D+00
vex2=0.1699080711134188D+00
f12(0)=1.d0
```

```
      f13(0)=1.d0
      f23(0)=1.d0
      bux12=r12*dexp(-vex1*r12)
      bux13=r13*dexp(-vex1*r13)
      bux23=r23*dexp(-vex2*r23)
      do 1 i=1, 8
         f12(i)=f12(i-1)*bux12
         f13(i)=f13(i-1)*bux13
         f23(i)=f23(i-1)*bux23
1     continue
      ener = 0.d0
      der12 = 0.d0
      der13 = 0.d0
      der23 = 0.d0
      do 2 l=1,106
         if (i4(l).eq.1) then
            aux=f12(i1(l))*f13(i3(l))*f23(i2(l))
            dux12=i1(l)*f12(i1(l)-1)*f13(i3(l))*f23(i2(l))
            dux13=i3(l)*f12(i1(l))*f13(i3(l)-1)*f23(i2(l))
            dux23=i2(l)*f12(i1(l))*f13(i3(l))*f23(i2(l)-1)
         else
            aux1=f12(i1(l))*f13(i3(l))
            aux2=f12(i3(l))*f13(i1(l))
            aux=(aux1+aux2)*f23(i2(l))
            dux23=(aux1+aux2)*i2(l)*f23(i2(l)-1)
            dux1=i1(l)*f12(i1(l)-1)*f13(i3(l))
            dux2=i3(l)*f12(i3(l)-1)*f13(i1(l))
            dux12=(dux1+dux2)*f23(i2(l))
            dux1=i3(l)*f12(i1(l))*f13(i3(l)-1)
            dux2=i1(l)*f12(i3(l))*f13(i1(l)-1)
            dux13=(dux1+dux2)*f23(i2(l))
         endif
         ener=ener+cf(l)*aux
         der12=der12+cf(l)*dux12
         der13=der13+cf(l)*dux13
         der23=der23+cf(l)*dux23
    2 continue
      der(1)=der12*(1.d0-vex1*r12)*dexp(-vex1*r12)
      der(2)=der13*(1.d0-vex1*r13)*dexp(-vex1*r13)
      der(3)=der23*(1.d0-vex2*r23)*dexp(-vex2*r23)
      return
      end
```

Fortran code 3. Analytical expression of the excited 1^1A" PES.

```
subroutine fit3d(r12,r13,r23,e,der)
      implicit real*8 (a-h,o-z)
      dimension der(3)
      call diat12(r12,e12,d12)
      call diat12(r13,e13,d13)
      call diat23(r23,e23,d23)
      call triabb(r12,r13,r23,e123,der)
      e=e12+e13+e23+e123
      der(1)=d12+der(1)
      der(2)=d13+der(2)
      der(3)=d23+der(3)
      return
      end
*************************************************************
********
      subroutine diat12(r,ener,der)
*************************************************************
********
*     This subroutine computes the energies of a diatomic
potential
*     fitted to    61 points
*     rms =      0.00415884 kcal/mol
*     emax =      0.01342953 kcal/mol
*************************************************************
********
      implicit real*8 (a-h,o-z)
      dimension cf( 11)
      data cf(  1)/0.1728681581542760D+01/
      data cf(  2)/-.8696895244194306D-01/
      data cf(  3)/-.1249140242092167D+01/
      data cf(  4)/0.7256887562139883D+01/
      data cf(  5)/-.3649313303125749D+01/
      data cf(  6)/-.2092960246381420D+03/
      data cf(  7)/0.1583838349649344D+04/
      data cf(  8)/-.6027970170568976D+04/
      data cf(  9)/0.1358740005165506D+05/
      data cf( 10)/-.1743446552915056D+05/
      data cf( 11)/0.1004855927106030D+05/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.9973983062075690D+00
      vex2=0.9430558414200502D+00
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2, 11
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
```

```
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
*************************************************************
********
      subroutine diat23(r,ener,der)
*************************************************************
********
*     This subroutine computes the energies of a diatomic
potential
*     fitted to    65 points
*     rms =      0.01590125 kcal/mol
*     emax =      0.05660348 kcal/mol
*************************************************************
********
      implicit real*8 (a-h,o-z)
      dimension cf( 11)
      data cf(  1)/0.1010010815772820D+01/
      data cf(  2)/-.7887699956696838D+00/
      data cf(  3)/0.4906159588439826D+01/
      data cf(  4)/-.6219863154243669D+02/
      data cf(  5)/0.6896895670668164D+03/
      data cf(  6)/-.5363366498220737D+04/
      data cf(  7)/0.2773950145917346D+05/
      data cf(  8)/-.9300358123786467D+05/
      data cf(  9)/0.1936313158084306D+06/
      data cf( 10)/-.2270595952691409D+06/
      data cf( 11)/0.1144414374786166D+06/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.9833832398267376D+00
      vex2=0.1631998058149494D+01
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2, 11
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
*************************************************************
      subroutine triabb(r12,r13,r23,ener,der)
*************************************************************
*     This subroutine computes the energies of a 3D PES
*     for the ABB system class fitted to  30497 points
*     rms =      0.26116725 kcal/mol
```

```
*     emax =      2.14840892 kcal/mol
*************************************************************
      implicit real*8(a-h,o-z)
      dimension i1(   55),i2(   55),i3(   55),i4(   55),cf(
55)
      dimension f12(0: 6),f13(0: 6),f23(0: 6)
      dimension der(3)
      data cf(  1)/0.2342649363134195D+00/
      data i1(  1)/ 1/,i2(  1)/ 1/,i3(  1)/ 0/,i4(  1)/ 2/
      data cf(  2)/-.1902316917762371D+00/
      data i1(  2)/ 1/,i2(  2)/ 0/,i3(  2)/ 1/,i4(  2)/ 1/
      data cf(  3)/0.1737364131123290D+01/
      data i1(  3)/ 1/,i2(  3)/ 1/,i3(  3)/ 1/,i4(  3)/ 1/
      data cf(  4)/-.2347585411761579D+01/
      data i1(  4)/ 2/,i2(  4)/ 1/,i3(  4)/ 0/,i4(  4)/ 2/
      data cf(  5)/0.4945380884828863D+00/
      data i1(  5)/ 2/,i2(  5)/ 0/,i3(  5)/ 1/,i4(  5)/ 2/
      data cf(  6)/-.8515994146754172D+00/
      data i1(  6)/ 0/,i2(  6)/ 2/,i3(  6)/ 1/,i4(  6)/ 2/
      data cf(  7)/0.3516059047583602D+01/
      data i1(  7)/ 2/,i2(  7)/ 1/,i3(  7)/ 1/,i4(  7)/ 2/
      data cf(  8)/-.1563268091733962D+02/
      data i1(  8)/ 1/,i2(  8)/ 2/,i3(  8)/ 1/,i4(  8)/ 1/
      data cf(  9)/0.1147342821857560D+02/
      data i1(  9)/ 2/,i2(  9)/ 2/,i3(  9)/ 0/,i4(  9)/ 2/
      data cf( 10)/-.6691375831332327D+01/
      data i1( 10)/ 2/,i2( 10)/ 0/,i3( 10)/ 2/,i4( 10)/ 1/
      data cf( 11)/0.3078228427848201D+01/
      data i1( 11)/ 3/,i2( 11)/ 1/,i3( 11)/ 0/,i4( 11)/ 2/
      data cf( 12)/0.1609668726285160D+01/
      data i1( 12)/ 3/,i2( 12)/ 0/,i3( 12)/ 1/,i4( 12)/ 2/
      data cf( 13)/0.4521149915913508D-01/
      data i1( 13)/ 0/,i2( 13)/ 3/,i3( 13)/ 1/,i4( 13)/ 2/
      data cf( 14)/0.6700886576956400D+01/
      data i1( 14)/ 2/,i2( 14)/ 2/,i3( 14)/ 1/,i4( 14)/ 2/
      data cf( 15)/0.1744910942940251D+02/
      data i1( 15)/ 2/,i2( 15)/ 1/,i3( 15)/ 2/,i4( 15)/ 1/
      data cf( 16)/-.2600169763196097D+02/
      data i1( 16)/ 3/,i2( 16)/ 1/,i3( 16)/ 1/,i4( 16)/ 2/
      data cf( 17)/0.2011225561278666D+02/
      data i1( 17)/ 1/,i2( 17)/ 3/,i3( 17)/ 1/,i4( 17)/ 1/
      data cf( 18)/-.1487637995020903D+02/
      data i1( 18)/ 3/,i2( 18)/ 2/,i3( 18)/ 0/,i4( 18)/ 2/
      data cf( 19)/0.7931589273652284D+01/
      data i1( 19)/ 3/,i2( 19)/ 0/,i3( 19)/ 2/,i4( 19)/ 2/
      data cf( 20)/-.1793292392595960D+02/
      data i1( 20)/ 0/,i2( 20)/ 3/,i3( 20)/ 2/,i4( 20)/ 2/
      data cf( 21)/0.3627782093842377D+01/
      data i1( 21)/ 4/,i2( 21)/ 1/,i3( 21)/ 0/,i4( 21)/ 2/
      data cf( 22)/-.3772197779122629D+01/
      data i1( 22)/ 4/,i2( 22)/ 0/,i3( 22)/ 1/,i4( 22)/ 2/
      data cf( 23)/0.8409798711966688D+01/
      data i1( 23)/ 0/,i2( 23)/ 4/,i3( 23)/ 1/,i4( 23)/ 2/
      data cf( 24)/-.9874613984960819D+02/
      data i1( 24)/ 2/,i2( 24)/ 2/,i3( 24)/ 2/,i4( 24)/ 1/
      data cf( 25)/0.7782573178329542D+02/
```

```
data i1( 25)/ 3/,i2( 25)/ 2/,i3( 25)/ 1/,i4( 25)/ 2/
data cf( 26)/-.4345781718353521D+00/
data i1( 26)/ 3/,i2( 26)/ 1/,i3( 26)/ 2/,i4( 26)/ 2/
data cf( 27)/-.1224877388313884D+02/
data i1( 27)/ 1/,i2( 27)/ 3/,i3( 27)/ 2/,i4( 27)/ 2/
data cf( 28)/0.1573181858784662D+02/
data i1( 28)/ 3/,i2( 28)/ 3/,i3( 28)/ 0/,i4( 28)/ 2/
data cf( 29)/-.2087867559670412D+02/
data i1( 29)/ 3/,i2( 29)/ 0/,i3( 29)/ 3/,i4( 29)/ 1/
data cf( 30)/0.1241826787951822D+02/
data i1( 30)/ 4/,i2( 30)/ 1/,i3( 30)/ 1/,i4( 30)/ 2/
data cf( 31)/0.2344246194649803D+01/
data i1( 31)/ 1/,i2( 31)/ 4/,i3( 31)/ 1/,i4( 31)/ 1/
data cf( 32)/-.1543865068570187D+02/
data i1( 32)/ 4/,i2( 32)/ 2/,i3( 32)/ 0/,i4( 32)/ 2/
data cf( 33)/0.6230011836151902D+01/
data i1( 33)/ 4/,i2( 33)/ 0/,i3( 33)/ 2/,i4( 33)/ 2/
data cf( 34)/0.9809965888773928D+01/
data i1( 34)/ 0/,i2( 34)/ 4/,i3( 34)/ 2/,i4( 34)/ 2/
data cf( 35)/0.1870290663385688D+00/
data i1( 35)/ 5/,i2( 35)/ 1/,i3( 35)/ 0/,i4( 35)/ 2/
data cf( 36)/-.6597258925557439D+00/
data i1( 36)/ 5/,i2( 36)/ 0/,i3( 36)/ 1/,i4( 36)/ 2/
data cf( 37)/-.1997049481377055D+02/
data i1( 37)/ 0/,i2( 37)/ 5/,i3( 37)/ 1/,i4( 37)/ 2/
data cf( 38)/-.7064058594209141D+01/
data i1( 38)/ 3/,i2( 38)/ 2/,i3( 38)/ 2/,i4( 38)/ 2/
data cf( 39)/0.1815625069759855D+03/
data i1( 39)/ 2/,i2( 39)/ 3/,i3( 39)/ 2/,i4( 39)/ 1/
data cf( 40)/-.6653337713170020D+02/
data i1( 40)/ 3/,i2( 40)/ 3/,i3( 40)/ 1/,i4( 40)/ 2/
data cf( 41)/0.2421960146054511D+02/
data i1( 41)/ 3/,i2( 41)/ 1/,i3( 41)/ 3/,i4( 41)/ 1/
data cf( 42)/-.2578854725097071D+02/
data i1( 42)/ 4/,i2( 42)/ 2/,i3( 42)/ 1/,i4( 42)/ 2/
data cf( 43)/-.1745109812961078D+02/
data i1( 43)/ 4/,i2( 43)/ 1/,i3( 43)/ 2/,i4( 43)/ 2/
data cf( 44)/-.5027308084126993D+02/
data i1( 44)/ 1/,i2( 44)/ 4/,i3( 44)/ 2/,i4( 44)/ 2/
data cf( 45)/-.1922084974272137D+02/
data i1( 45)/ 4/,i2( 45)/ 3/,i3( 45)/ 0/,i4( 45)/ 2/
data cf( 46)/0.7489127176309867D+01/
data i1( 46)/ 4/,i2( 46)/ 0/,i3( 46)/ 3/,i4( 46)/ 2/
data cf( 47)/0.4365038664708807D+02/
data i1( 47)/ 0/,i2( 47)/ 4/,i3( 47)/ 3/,i4( 47)/ 2/
data cf( 48)/0.8779147203698517D+01/
data i1( 48)/ 5/,i2( 48)/ 1/,i3( 48)/ 1/,i4( 48)/ 2/
data cf( 49)/0.3236104687500785D+02/
data i1( 49)/ 1/,i2( 49)/ 5/,i3( 49)/ 1/,i4( 49)/ 1/
data cf( 50)/0.2140927901278137D+02/
data i1( 50)/ 5/,i2( 50)/ 2/,i3( 50)/ 0/,i4( 50)/ 2/
data cf( 51)/-.8361001559736217D+01/
data i1( 51)/ 5/,i2( 51)/ 0/,i3( 51)/ 2/,i4( 51)/ 2/
data cf( 52)/-.2025277884541815D+02/
data i1( 52)/ 0/,i2( 52)/ 5/,i3( 52)/ 2/,i4( 52)/ 2/
data cf( 53)/-.6048922520343221D+01/
```

```
      data i1( 53)/ 6/,i2( 53)/ 1/,i3( 53)/ 0/,i4( 53)/ 2/
      data cf( 54)/0.2996716937672140D+01/
      data i1( 54)/ 6/,i2( 54)/ 0/,i3( 54)/ 1/,i4( 54)/ 2/
      data cf( 55)/0.1356567023776261D+02/
      data i1( 55)/ 0/,i2( 55)/ 6/,i3( 55)/ 1/,i4( 55)/ 2/
      vex1=0.4559920117533886D+00
      vex2=0.7144847301068902D+00
      f12(0)=1.d0
      f13(0)=1.d0
      f23(0)=1.d0
      bux12=r12*dexp(-vex1*r12)
      bux13=r13*dexp(-vex1*r13)
      bux23=r23*dexp(-vex2*r23)
      do 1 i=1, 6
         f12(i)=f12(i-1)*bux12
         f13(i)=f13(i-1)*bux13
         f23(i)=f23(i-1)*bux23
1     continue
      ener = 0.d0
      der12 = 0.d0
      der13 = 0.d0
      der23 = 0.d0
      do 2 l=1, 55
         if (i4(l).eq.1) then
            aux=f12(i1(l))*f13(i3(l))*f23(i2(l))
            dux12=i1(l)*f12(i1(l)-1)*f13(i3(l))*f23(i2(l))
            dux13=i3(l)*f12(i1(l))*f13(i3(l)-1)*f23(i2(l))
            dux23=i2(l)*f12(i1(l))*f13(i3(l))*f23(i2(l)-1)
         else
            aux1=f12(i1(l))*f13(i3(l))
            aux2=f12(i3(l))*f13(i1(l))
            aux=(aux1+aux2)*f23(i2(l))
            dux23=(aux1+aux2)*i2(l)*f23(i2(l)-1)
            dux1=i1(l)*f12(i1(l)-1)*f13(i3(l))
            dux2=i3(l)*f12(i3(l)-1)*f13(i1(l))
            dux12=(dux1+dux2)*f23(i2(l))
            dux1=i3(l)*f12(i1(l))*f13(i3(l)-1)
            dux2=i1(l)*f12(i3(l))*f13(i1(l)-1)
            dux13=(dux1+dux2)*f23(i2(l))
         endif
         ener=ener+cf(l)*aux
         der12=der12+cf(l)*dux12
         der13=der13+cf(l)*dux13
         der23=der23+cf(l)*dux23
    2 continue
      der(1)=der12*(1.d0-vex1*r12)*dexp(-vex1*r12)
      der(2)=der13*(1.d0-vex1*r13)*dexp(-vex1*r13)
      der(3)=der23*(1.d0-vex2*r23)*dexp(-vex2*r23)
      return
      end
```

Fortran code 4. Analytical expression of the excited $1^3A'$ PES.

```
 subroutine fit3d(r12,r13,r23,e,der)
       implicit real*8 (a-h,o-z)
       dimension der(3)
       call diat12(r12,e12,d12)
       call diat12(r13,e13,d13)
       call diat23(r23,e23,d23)
       call triabb(r12,r13,r23,e123,der)
       e=e12+e13+e23+e123
       der(1)=d12+der(1)
       der(2)=d13+der(2)
       der(3)=d23+der(3)
       return
       end
*****************************************************************
********
       subroutine diat12(r,ener,der)
*****************************************************************
********
*      This subroutine computes the energies of a diatomic
potential
*      fitted to    57 points
*      rms =      0.00602335 kcal/mol
*      emax =      0.01563994 kcal/mol
*****************************************************************
********
       implicit real*8 (a-h,o-z)
       dimension cf(  9)
       data cf(  1)/0.1538173115866930D+01/
       data cf(  2)/-.1672665770259515D-01/
       data cf(  3)/-.6216496440374958D-01/
       data cf(  4)/-.1265761585408977D+01/
       data cf(  5)/0.6390811027765745D+01/
       data cf(  6)/-.2030701278242597D+02/
       data cf(  7)/0.3775784829665356D+02/
       data cf(  8)/-.3766808177685533D+02/
       data cf(  9)/0.1525302873820288D+02/
       e0=0.0000000000000000D+00
       der=0.d0
       vex1=0.6318438807074429D+00
       vex2=0.7405123792895593D+00
       aux = 1.d0/r
       bux = dexp(-vex2*r)*aux
       cux = dexp(-vex1*r)
       ener=e0+cf(1)*bux
       dux=1.d0
       eux=r*cux
       do 1 i=2,  9
          der=der+(i-1)*cf(i)*dux
          dux=dux*eux
          ener=ener+cf(i)*dux
    1 continue
```

```
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
*************************************************************
********
      subroutine diat23(r,ener,der)
*************************************************************
********
*     This subroutine computes the energies of a diatomic
potential
*     fitted to    67 points
*     rms =      0.01731213 kcal/mol
*     emax =      0.04233003 kcal/mol
*************************************************************
********
      implicit real*8 (a-h,o-z)
      dimension cf( 11)
      data cf(  1)/0.1337847086959146D+01/
      data cf(  2)/0.2585800665293710D+01/
      data cf(  3)/-.4004748068631668D+02/
      data cf(  4)/0.1343733522297093D+04/
      data cf(  5)/-.3170516869807047D+05/
      data cf(  6)/0.4885923727131493D+06/
      data cf(  7)/-.4918701586711237D+07/
      data cf(  8)/0.3203933330855191D+08/
      data cf(  9)/-.1299193151010496D+09/
      data cf( 10)/0.2978127229925779D+09/
      data cf( 11)/-.2944933154669650D+09/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.1832053626039320D+01
      vex2=0.2386896311658729D+01
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2, 11
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
*************************************************************
      subroutine triabb(r12,r13,r23,ener,der)
*************************************************************
*     This subroutine computes the energies of a 3D PES
*     for the ABB system class fitted to  40230 points
*     rms =      0.37363045 kcal/mol
*     emax =      2.43593629 kcal/mol
*************************************************************
```

```
      implicit real*8(a-h,o-z)
      dimension i1(   78),i2(   78),i3(   78),i4(   78),cf(
78)
      dimension f12(0: 7),f13(0: 7),f23(0: 7)
      dimension der(3)
      data cf(  1)/0.1593465762284637D+01/
      data i1(  1)/ 1/,i2(  1)/ 1/,i3(  1)/ 0/,i4(  1)/ 2/
      data cf(  2)/0.1380539088104646D+00/
      data i1(  2)/ 1/,i2(  2)/ 0/,i3(  2)/ 1/,i4(  2)/ 1/
      data cf(  3)/0.1086175813726766D+03/
      data i1(  3)/ 1/,i2(  3)/ 1/,i3(  3)/ 1/,i4(  3)/ 1/
      data cf(  4)/-.6005161505064049D+02/
      data i1(  4)/ 2/,i2(  4)/ 1/,i3(  4)/ 0/,i4(  4)/ 2/
      data cf(  5)/-.8367227421423601D+00/
      data i1(  5)/ 2/,i2(  5)/ 0/,i3(  5)/ 1/,i4(  5)/ 2/
      data cf(  6)/-.9538199066014516D+02/
      data i1(  6)/ 0/,i2(  6)/ 2/,i3(  6)/ 1/,i4(  6)/ 2/
      data cf(  7)/-.2579468742331817D+03/
      data i1(  7)/ 2/,i2(  7)/ 1/,i3(  7)/ 1/,i4(  7)/ 2/
      data cf(  8)/-.1990641075141963D+04/
      data i1(  8)/ 1/,i2(  8)/ 2/,i3(  8)/ 1/,i4(  8)/ 1/
      data cf(  9)/0.1075008272369665D+04/
      data i1(  9)/ 2/,i2(  9)/ 2/,i3(  9)/ 0/,i4(  9)/ 2/
      data cf( 10)/0.3920458090925745D+01/
      data i1( 10)/ 2/,i2( 10)/ 0/,i3( 10)/ 2/,i4( 10)/ 1/
      data cf( 11)/0.2882068031286211D+03/
      data i1( 11)/ 3/,i2( 11)/ 1/,i3( 11)/ 0/,i4( 11)/ 2/
      data cf( 12)/0.2086371098015231D+01/
      data i1( 12)/ 3/,i2( 12)/ 0/,i3( 12)/ 1/,i4( 12)/ 2/
      data cf( 13)/0.2692508022386851D+04/
      data i1( 13)/ 0/,i2( 13)/ 3/,i3( 13)/ 1/,i4( 13)/ 2/
      data cf( 14)/-.1336310592400417D+04/
      data i1( 14)/ 2/,i2( 14)/ 2/,i3( 14)/ 1/,i4( 14)/ 2/
      data cf( 15)/0.3910374962064418D+04/
      data i1( 15)/ 2/,i2( 15)/ 1/,i3( 15)/ 2/,i4( 15)/ 1/
      data cf( 16)/-.1562118277230409D+04/
      data i1( 16)/ 3/,i2( 16)/ 1/,i3( 16)/ 1/,i4( 16)/ 2/
      data cf( 17)/0.7360605877000864D+05/
      data i1( 17)/ 1/,i2( 17)/ 3/,i3( 17)/ 1/,i4( 17)/ 1/
      data cf( 18)/0.1436787019377376D+04/
      data i1( 18)/ 3/,i2( 18)/ 2/,i3( 18)/ 0/,i4( 18)/ 2/
      data cf( 19)/-.7733131105498105D+01/
      data i1( 19)/ 3/,i2( 19)/ 0/,i3( 19)/ 2/,i4( 19)/ 2/
      data cf( 20)/-.4055086002660169D+05/
      data i1( 20)/ 0/,i2( 20)/ 3/,i3( 20)/ 2/,i4( 20)/ 2/
      data cf( 21)/-.4759353945880831D+03/
      data i1( 21)/ 4/,i2( 21)/ 1/,i3( 21)/ 0/,i4( 21)/ 2/
      data cf( 22)/-.2047989311576056D+01/
      data i1( 22)/ 4/,i2( 22)/ 0/,i3( 22)/ 1/,i4( 22)/ 2/
      data cf( 23)/-.3093224073182304D+05/
      data i1( 23)/ 0/,i2( 23)/ 4/,i3( 23)/ 1/,i4( 23)/ 2/
      data cf( 24)/-.1431290973063213D+05/
      data i1( 24)/ 2/,i2( 24)/ 2/,i3( 24)/ 2/,i4( 24)/ 1/
      data cf( 25)/0.1530265864709914D+05/
      data i1( 25)/ 3/,i2( 25)/ 2/,i3( 25)/ 1/,i4( 25)/ 2/
      data cf( 26)/-.3606215848767661D+04/
```

```
data i1( 26)/ 3/,i2( 26)/ 1/,i3( 26)/ 2/,i4( 26)/ 2/
data cf( 27)/-.2613763389737555D+05/
data i1( 27)/ 1/,i2( 27)/ 3/,i3( 27)/ 2/,i4( 27)/ 2/
data cf( 28)/0.2425502714013408D+05/
data i1( 28)/ 3/,i2( 28)/ 3/,i3( 28)/ 0/,i4( 28)/ 2/
data cf( 29)/0.1200534790584008D+02/
data i1( 29)/ 3/,i2( 29)/ 0/,i3( 29)/ 3/,i4( 29)/ 1/
data cf( 30)/0.3137151901926974D+04/
data i1( 30)/ 4/,i2( 30)/ 1/,i3( 30)/ 1/,i4( 30)/ 2/
data cf( 31)/-.5873759326100443D+06/
data i1( 31)/ 1/,i2( 31)/ 4/,i3( 31)/ 1/,i4( 31)/ 1/
data cf( 32)/-.7761538092995199D+04/
data i1( 32)/ 4/,i2( 32)/ 2/,i3( 32)/ 0/,i4( 32)/ 2/
data cf( 33)/0.6262046515077513D+01/
data i1( 33)/ 4/,i2( 33)/ 0/,i3( 33)/ 2/,i4( 33)/ 2/
data cf( 34)/0.3414222239321112D+06/
data i1( 34)/ 0/,i2( 34)/ 4/,i3( 34)/ 2/,i4( 34)/ 2/
data cf( 35)/0.5444578246527525D+03/
data i1( 35)/ 5/,i2( 35)/ 1/,i3( 35)/ 0/,i4( 35)/ 2/
data cf( 36)/0.1246835650882357D+00/
data i1( 36)/ 5/,i2( 36)/ 0/,i3( 36)/ 1/,i4( 36)/ 2/
data cf( 37)/0.1528994259211896D+06/
data i1( 37)/ 0/,i2( 37)/ 5/,i3( 37)/ 1/,i4( 37)/ 2/
data cf( 38)/0.2210418489128586D+04/
data i1( 38)/ 3/,i2( 38)/ 2/,i3( 38)/ 2/,i4( 38)/ 2/
data cf( 39)/0.6347549389035141D+06/
data i1( 39)/ 2/,i2( 39)/ 3/,i3( 39)/ 2/,i4( 39)/ 1/
data cf( 40)/-.4254425906536806D+06/
data i1( 40)/ 3/,i2( 40)/ 3/,i3( 40)/ 1/,i4( 40)/ 2/
data cf( 41)/0.1049646422371278D+05/
data i1( 41)/ 3/,i2( 41)/ 1/,i3( 41)/ 3/,i4( 41)/ 1/
data cf( 42)/-.9050466544991188D+04/
data i1( 42)/ 4/,i2( 42)/ 2/,i3( 42)/ 1/,i4( 42)/ 2/
data cf( 43)/-.4065825329483570D+04/
data i1( 43)/ 4/,i2( 43)/ 1/,i3( 43)/ 2/,i4( 43)/ 2/
data cf( 44)/0.1996686499712976D+06/
data i1( 44)/ 1/,i2( 44)/ 4/,i3( 44)/ 2/,i4( 44)/ 2/
data cf( 45)/0.1154313339123439D+06/
data i1( 45)/ 4/,i2( 45)/ 3/,i3( 45)/ 0/,i4( 45)/ 2/
data cf( 46)/-.7939490372606879D+01/
data i1( 46)/ 4/,i2( 46)/ 0/,i3( 46)/ 3/,i4( 46)/ 2/
data cf( 47)/-.2230505065362797D+06/
data i1( 47)/ 0/,i2( 47)/ 4/,i3( 47)/ 3/,i4( 47)/ 2/
data cf( 48)/-.5323744879553769D+03/
data i1( 48)/ 5/,i2( 48)/ 1/,i3( 48)/ 1/,i4( 48)/ 2/
data cf( 49)/0.1581722032440149D+07/
data i1( 49)/ 1/,i2( 49)/ 5/,i3( 49)/ 1/,i4( 49)/ 1/
data cf( 50)/0.5471325567116290D+04/
data i1( 50)/ 5/,i2( 50)/ 2/,i3( 50)/ 0/,i4( 50)/ 2/
data cf( 51)/-.4816178245462967D+00/
data i1( 51)/ 5/,i2( 51)/ 0/,i3( 51)/ 2/,i4( 51)/ 2/
data cf( 52)/-.9690698851641440D+06/
data i1( 52)/ 0/,i2( 52)/ 5/,i3( 52)/ 2/,i4( 52)/ 2/
data cf( 53)/-.6205771199970242D+03/
data i1( 53)/ 6/,i2( 53)/ 1/,i3( 53)/ 0/,i4( 53)/ 2/
data cf( 54)/0.7282616294734210D+00/
```

```
data i1( 54)/ 6/,i2( 54)/ 0/,i3( 54)/ 1/,i4( 54)/ 2/
data cf( 55)/-.3286670281781343D+06/
data i1( 55)/ 0/,i2( 55)/ 6/,i3( 55)/ 1/,i4( 55)/ 2/
data cf( 56)/-.9203218721046858D+05/
data i1( 56)/ 3/,i2( 56)/ 3/,i3( 56)/ 2/,i4( 56)/ 2/
data cf( 57)/0.3870686742263555D+06/
data i1( 57)/ 3/,i2( 57)/ 2/,i3( 57)/ 3/,i4( 57)/ 1/
data cf( 58)/-.2882353344999062D+06/
data i1( 58)/ 4/,i2( 58)/ 2/,i3( 58)/ 2/,i4( 58)/ 2/
data cf( 59)/-.1785668505249989D+07/
data i1( 59)/ 2/,i2( 59)/ 4/,i3( 59)/ 2/,i4( 59)/ 1/
data cf( 60)/0.1413869011541432D+06/
data i1( 60)/ 4/,i2( 60)/ 3/,i3( 60)/ 1/,i4( 60)/ 2/
data cf( 61)/-.2055986605929290D+04/
data i1( 61)/ 4/,i2( 61)/ 1/,i3( 61)/ 3/,i4( 61)/ 2/
data cf( 62)/0.1175186161730540D+07/
data i1( 62)/ 1/,i2( 62)/ 4/,i3( 62)/ 3/,i4( 62)/ 2/
data cf( 63)/-.2674600624254684D+06/
data i1( 63)/ 4/,i2( 63)/ 4/,i3( 63)/ 0/,i4( 63)/ 2/
data cf( 64)/0.1273608702387414D+01/
data i1( 64)/ 4/,i2( 64)/ 0/,i3( 64)/ 4/,i4( 64)/ 1/
data cf( 65)/0.1148665650951955D+06/
data i1( 65)/ 5/,i2( 65)/ 2/,i3( 65)/ 1/,i4( 65)/ 2/
data cf( 66)/0.2040158035407468D+04/
data i1( 66)/ 5/,i2( 66)/ 1/,i3( 66)/ 2/,i4( 66)/ 2/
data cf( 67)/-.4200365505504206D+06/
data i1( 67)/ 1/,i2( 67)/ 5/,i3( 67)/ 2/,i4( 67)/ 2/
data cf( 68)/-.5505805614885832D+05/
data i1( 68)/ 5/,i2( 68)/ 3/,i3( 68)/ 0/,i4( 68)/ 2/
data cf( 69)/0.2760769720315645D+01/
data i1( 69)/ 5/,i2( 69)/ 0/,i3( 69)/ 3/,i4( 69)/ 2/
data cf( 70)/0.4197750118947557D+06/
data i1( 70)/ 0/,i2( 70)/ 5/,i3( 70)/ 3/,i4( 70)/ 2/
data cf( 71)/-.2860370085701244D+03/
data i1( 71)/ 6/,i2( 71)/ 1/,i3( 71)/ 1/,i4( 71)/ 2/
data cf( 72)/-.8241155591249131D+06/
data i1( 72)/ 1/,i2( 72)/ 6/,i3( 72)/ 1/,i4( 72)/ 1/
data cf( 73)/-.1913839389343370D+05/
data i1( 73)/ 6/,i2( 73)/ 2/,i3( 73)/ 0/,i4( 73)/ 2/
data cf( 74)/-.1502896313906952D+01/
data i1( 74)/ 6/,i2( 74)/ 0/,i3( 74)/ 2/,i4( 74)/ 2/
data cf( 75)/0.7280936741006975D+06/
data i1( 75)/ 0/,i2( 75)/ 6/,i3( 75)/ 2/,i4( 75)/ 2/
data cf( 76)/0.2464149454186640D+03/
data i1( 76)/ 7/,i2( 76)/ 1/,i3( 76)/ 0/,i4( 76)/ 2/
data cf( 77)/-.1291391050791530D+00/
data i1( 77)/ 7/,i2( 77)/ 0/,i3( 77)/ 1/,i4( 77)/ 2/
data cf( 78)/0.1656446817456111D+06/
data i1( 78)/ 0/,i2( 78)/ 7/,i3( 78)/ 1/,i4( 78)/ 2/
vex1=0.4437106205418788D+00
vex2=0.1826295667676822D+01
f12(0)=1.d0
f13(0)=1.d0
f23(0)=1.d0
bux12=r12*dexp(-vex1*r12)
bux13=r13*dexp(-vex1*r13)
```

```
      bux23=r23*dexp(-vex2*r23)
      do 1 i=1, 7
         f12(i)=f12(i-1)*bux12
         f13(i)=f13(i-1)*bux13
         f23(i)=f23(i-1)*bux23
1     continue
      ener = 0.d0
      der12 = 0.d0
      der13 = 0.d0
      der23 = 0.d0
      do 2 l=1, 78
         if (i4(l).eq.1) then
            aux=f12(i1(l))*f13(i3(l))*f23(i2(l))
            dux12=i1(l)*f12(i1(l)-1)*f13(i3(l))*f23(i2(l))
            dux13=i3(l)*f12(i1(l))*f13(i3(l)-1)*f23(i2(l))
            dux23=i2(l)*f12(i1(l))*f13(i3(l))*f23(i2(l)-1)
         else
            aux1=f12(i1(l))*f13(i3(l))
            aux2=f12(i3(l))*f13(i1(l))
            aux=(aux1+aux2)*f23(i2(l))
            dux23=(aux1+aux2)*i2(l)*f23(i2(l)-1)
            dux1=i1(l)*f12(i1(l)-1)*f13(i3(l))
            dux2=i3(l)*f12(i3(l)-1)*f13(i1(l))
            dux12=(dux1+dux2)*f23(i2(l))
            dux1=i3(l)*f12(i1(l))*f13(i3(l)-1)
            dux2=i1(l)*f12(i3(l))*f13(i1(l)-1)
            dux13=(dux1+dux2)*f23(i2(l))
         endif
         ener=ener+cf(l)*aux
         der12=der12+cf(l)*dux12
         der13=der13+cf(l)*dux13
         der23=der23+cf(l)*dux23
    2 continue
      der(1)=der12*(1.d0-vex1*r12)*dexp(-vex1*r12)
      der(2)=der13*(1.d0-vex1*r13)*dexp(-vex1*r13)
      der(3)=der23*(1.d0-vex2*r23)*dexp(-vex2*r23)
      return
      end
```

Fortran code 5. Analytical expression of the excited 2^3A' PES.

```
  subroutine fit3d(r12,r13,r23,e,der)
      implicit real*8 (a-h,o-z)
      dimension der(3)
      call diat12(r12,e12,d12)
      call diat12(r13,e13,d13)
      call diat23(r23,e23,d23)
      call triabb(r12,r13,r23,e123,der)
      e=e12+e13+e23+e123
      der(1)=d12+der(1)
      der(2)=d13+der(2)
      der(3)=d23+der(3)
      return
      end
!*************************************************************
*********
      subroutine diat12(r,ener,der)
!*************************************************************
*********
!*     This subroutine computes the energies of a diatomic
potential
!*     fitted to    59 points
!*     rms =      0.11209395 kcal/mol
!*     emax =      0.23877383 kcal/mol
!*************************************************************
*********
      implicit real*8 (a-h,o-z)
      dimension cf( 11)
      data cf(  1)/0.1619944524011548D+01/
      data cf(  2)/-.6469801592030440D+00/
      data cf(  3)/-.1951243216250550D+02/
      data cf(  4)/0.7453430357212217D+03/
      data cf(  5)/-.1565854316752900D+05/
      data cf(  6)/0.2065658489679768D+06/
      data cf(  7)/-.1746990835027075D+07/
      data cf(  8)/0.9419126493665289D+07/
      data cf(  9)/-.3122979702528037D+08/
      data cf( 10)/0.5793801158781502D+08/
      data cf( 11)/-.4600057711331255D+08/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.1364941401920840D+01
      vex2=0.6764450547918991D+00
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2, 11
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
```

```
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
!*************************************************************
*********
       subroutine diat23(r,ener,der)
!*************************************************************
*********
!*     This subroutine computes the energies of a diatomic
potential
!*     fitted to    72 points
!*     rms =      0.01501997 kcal/mol
!*     emax =      0.05458411 kcal/mol
!*************************************************************
*********
      implicit real*8 (a-h,o-z)
      dimension cf( 11)
      data cf(  1)/0.1022297382480432D+01/
      data cf(  2)/-.7672738217416850D+00/
      data cf(  3)/0.4264477545997991D+01/
      data cf(  4)/-.4708724852634353D+02/
      data cf(  5)/0.4641581157278750D+03/
      data cf(  6)/-.3259025457886833D+04/
      data cf(  7)/0.1533491920089703D+05/
      data cf(  8)/-.4691860547744167D+05/
      data cf(  9)/0.8921625418895467D+05/
      data cf( 10)/-.9551643860478746D+05/
      data cf( 11)/0.4391295772917960D+05/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.9798843955181915D+00
      vex2=0.1654128603889533D+01
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2, 11
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
!*************************************************************
       subroutine triabb(r12,r13,r23,ener,der)
!*************************************************************
!*     This subroutine computes the energies of a 3D PES
!*     for the ABB system class fitted to  49478 points
!*     rms =      1.74367898 kcal/mol
```

```
!*     emax =     24.36870975 kcal/mol
!*******************************************************
      implicit real*8(a-h,o-z)
      dimension i1(   55),i2(   55),i3(   55),i4(   55),cf(
55)
      dimension f12(0: 6),f13(0: 6),f23(0: 6)
      dimension der(3)
      data cf(  1)/0.5232552146888330D+00/
      data i1(  1)/ 1/,i2(  1)/ 1/,i3(  1)/ 0/,i4(  1)/ 2/
      data cf(  2)/-.2132564933936615D+01/
      data i1(  2)/ 1/,i2(  2)/ 0/,i3(  2)/ 1/,i4(  2)/ 1/
      data cf(  3)/0.2045443743872815D+02/
      data i1(  3)/ 1/,i2(  3)/ 1/,i3(  3)/ 1/,i4(  3)/ 1/
      data cf(  4)/-.6908645545976043D+01/
      data i1(  4)/ 2/,i2(  4)/ 1/,i3(  4)/ 0/,i4(  4)/ 2/
      data cf(  5)/0.7609375456584592D+01/
      data i1(  5)/ 2/,i2(  5)/ 0/,i3(  5)/ 1/,i4(  5)/ 2/
      data cf(  6)/-.1370415006709562D+02/
      data i1(  6)/ 0/,i2(  6)/ 2/,i3(  6)/ 1/,i4(  6)/ 2/
      data cf(  7)/-.1794576207481303D+02/
      data i1(  7)/ 2/,i2(  7)/ 1/,i3(  7)/ 1/,i4(  7)/ 2/
      data cf(  8)/-.2224118551251306D+03/
      data i1(  8)/ 1/,i2(  8)/ 2/,i3(  8)/ 1/,i4(  8)/ 1/
      data cf(  9)/0.9660061038804263D+02/
      data i1(  9)/ 2/,i2(  9)/ 2/,i3(  9)/ 0/,i4(  9)/ 2/
      data cf( 10)/-.1190490751503718D+02/
      data i1( 10)/ 2/,i2( 10)/ 0/,i3( 10)/ 2/,i4( 10)/ 1/
      data cf( 11)/-.8712287579247945D+00/
      data i1( 11)/ 3/,i2( 11)/ 1/,i3( 11)/ 0/,i4( 11)/ 2/
      data cf( 12)/-.1816337192585021D+02/
      data i1( 12)/ 3/,i2( 12)/ 0/,i3( 12)/ 1/,i4( 12)/ 2/
      data cf( 13)/0.1442031705863976D+03/
      data i1( 13)/ 0/,i2( 13)/ 3/,i3( 13)/ 1/,i4( 13)/ 2/
      data cf( 14)/0.2508259215339758D+01/
      data i1( 14)/ 2/,i2( 14)/ 2/,i3( 14)/ 1/,i4( 14)/ 2/
      data cf( 15)/0.5617585522567757D+03/
      data i1( 15)/ 2/,i2( 15)/ 1/,i3( 15)/ 2/,i4( 15)/ 1/
      data cf( 16)/-.3480463034301471D+03/
      data i1( 16)/ 3/,i2( 16)/ 1/,i3( 16)/ 1/,i4( 16)/ 2/
      data cf( 17)/0.1506039256688979D+04/
      data i1( 17)/ 1/,i2( 17)/ 3/,i3( 17)/ 1/,i4( 17)/ 1/
      data cf( 18)/0.5610334915719325D+02/
      data i1( 18)/ 3/,i2( 18)/ 2/,i3( 18)/ 0/,i4( 18)/ 2/
      data cf( 19)/0.8043831322760178D+01/
      data i1( 19)/ 3/,i2( 19)/ 0/,i3( 19)/ 2/,i4( 19)/ 2/
      data cf( 20)/-.7557649494536001D+03/
      data i1( 20)/ 0/,i2( 20)/ 3/,i3( 20)/ 2/,i4( 20)/ 2/
      data cf( 21)/0.1032309532000700D+03/
      data i1( 21)/ 4/,i2( 21)/ 1/,i3( 21)/ 0/,i4( 21)/ 2/
      data cf( 22)/0.3205644178267772D+02/
      data i1( 22)/ 4/,i2( 22)/ 0/,i3( 22)/ 1/,i4( 22)/ 2/
      data cf( 23)/-.6696710334284481D+03/
      data i1( 23)/ 0/,i2( 23)/ 4/,i3( 23)/ 1/,i4( 23)/ 2/
      data cf( 24)/-.2306731096968502D+04/
      data i1( 24)/ 2/,i2( 24)/ 2/,i3( 24)/ 2/,i4( 24)/ 1/
      data cf( 25)/0.1546621552661791D+04/
```

```
data i1( 25)/ 3/,i2( 25)/ 2/,i3( 25)/ 1/,i4( 25)/ 2/
data cf( 26)/-.3386214366580460D+03/
data i1( 26)/ 3/,i2( 26)/ 1/,i3( 26)/ 2/,i4( 26)/ 2/
data cf( 27)/0.5316523681797708D+02/
data i1( 27)/ 1/,i2( 27)/ 3/,i3( 27)/ 2/,i4( 27)/ 2/
data cf( 28)/-.1767078512137205D+03/
data i1( 28)/ 3/,i2( 28)/ 3/,i3( 28)/ 0/,i4( 28)/ 2/
data cf( 29)/-.6586162862171294D+01/
data i1( 29)/ 3/,i2( 29)/ 0/,i3( 29)/ 3/,i4( 29)/ 1/
data cf( 30)/0.4845255701730322D+03/
data i1( 30)/ 4/,i2( 30)/ 1/,i3( 30)/ 1/,i4( 30)/ 2/
data cf( 31)/-.5539778329748754D+04/
data i1( 31)/ 1/,i2( 31)/ 4/,i3( 31)/ 1/,i4( 31)/ 1/
data cf( 32)/-.4664277361965960D+03/
data i1( 32)/ 4/,i2( 32)/ 2/,i3( 32)/ 0/,i4( 32)/ 2/
data cf( 33)/-.3007657670842094D+00/
data i1( 33)/ 4/,i2( 33)/ 0/,i3( 33)/ 2/,i4( 33)/ 2/
data cf( 34)/0.3155434521205943D+04/
data i1( 34)/ 0/,i2( 34)/ 4/,i3( 34)/ 2/,i4( 34)/ 2/
data cf( 35)/-.1734768098511554D+03/
data i1( 35)/ 5/,i2( 35)/ 1/,i3( 35)/ 0/,i4( 35)/ 2/
data cf( 36)/-.3009097123735941D+02/
data i1( 36)/ 5/,i2( 36)/ 0/,i3( 36)/ 1/,i4( 36)/ 2/
data cf( 37)/0.1109921345503035D+04/
data i1( 37)/ 0/,i2( 37)/ 5/,i3( 37)/ 1/,i4( 37)/ 2/
data cf( 38)/0.5383565023020273D+03/
data i1( 38)/ 3/,i2( 38)/ 2/,i3( 38)/ 2/,i4( 38)/ 2/
data cf( 39)/0.4117941364127629D+04/
data i1( 39)/ 2/,i2( 39)/ 3/,i3( 39)/ 2/,i4( 39)/ 1/
data cf( 40)/-.3326947746341464D+04/
data i1( 40)/ 3/,i2( 40)/ 3/,i3( 40)/ 1/,i4( 40)/ 2/
data cf( 41)/-.6540938393755469D+03/
data i1( 41)/ 3/,i2( 41)/ 1/,i3( 41)/ 3/,i4( 41)/ 1/
data cf( 42)/-.6773735908002590D+03/
data i1( 42)/ 4/,i2( 42)/ 2/,i3( 42)/ 1/,i4( 42)/ 2/
data cf( 43)/0.6249265063642952D+03/
data i1( 43)/ 4/,i2( 43)/ 1/,i3( 43)/ 2/,i4( 43)/ 2/
data cf( 44)/0.1195094570966182D+04/
data i1( 44)/ 1/,i2( 44)/ 4/,i3( 44)/ 2/,i4( 44)/ 2/
data cf( 45)/0.1244824287782791D+04/
data i1( 45)/ 4/,i2( 45)/ 3/,i3( 45)/ 0/,i4( 45)/ 2/
data cf( 46)/0.1634022377657989D+01/
data i1( 46)/ 4/,i2( 46)/ 0/,i3( 46)/ 3/,i4( 46)/ 2/
data cf( 47)/-.9154233058144278D+03/
data i1( 47)/ 0/,i2( 47)/ 4/,i3( 47)/ 3/,i4( 47)/ 2/
data cf( 48)/-.4163481478825207D+03/
data i1( 48)/ 5/,i2( 48)/ 1/,i3( 48)/ 1/,i4( 48)/ 2/
data cf( 49)/0.3704490069254743D+04/
data i1( 49)/ 1/,i2( 49)/ 5/,i3( 49)/ 1/,i4( 49)/ 1/
data cf( 50)/0.1849344384270341D+03/
data i1( 50)/ 5/,i2( 50)/ 2/,i3( 50)/ 0/,i4( 50)/ 2/
data cf( 51)/-.7023057027022565D+00/
data i1( 51)/ 5/,i2( 51)/ 0/,i3( 51)/ 2/,i4( 51)/ 2/
data cf( 52)/-.2919012209123642D+04/
data i1( 52)/ 0/,i2( 52)/ 5/,i3( 52)/ 2/,i4( 52)/ 2/
data cf( 53)/0.1245799420454103D+03/
```

```
      data i1( 53)/ 6/,i2( 53)/ 1/,i3( 53)/ 0/,i4( 53)/ 2/
      data cf( 54)/0.1044087866747009D+02/
      data i1( 54)/ 6/,i2( 54)/ 0/,i3( 54)/ 1/,i4( 54)/ 2/
      data cf( 55)/-.6395813155240017D+02/
      data i1( 55)/ 0/,i2( 55)/ 6/,i3( 55)/ 1/,i4( 55)/ 2/
      vex1=0.4249906679652683D+00
      vex2=0.1124660677681497D+01
      f12(0)=1.d0
      f13(0)=1.d0
      f23(0)=1.d0
      bux12=r12*dexp(-vex1*r12)
      bux13=r13*dexp(-vex1*r13)
      bux23=r23*dexp(-vex2*r23)
      do 1 i=1, 6
         f12(i)=f12(i-1)*bux12
         f13(i)=f13(i-1)*bux13
         f23(i)=f23(i-1)*bux23
1     continue
      ener = 0.d0
      der12 = 0.d0
      der13 = 0.d0
      der23 = 0.d0
      do 2 l=1, 55
         if (i4(l).eq.1) then
            aux=f12(i1(l))*f13(i3(l))*f23(i2(l))
            dux12=i1(l)*f12(i1(l)-1)*f13(i3(l))*f23(i2(l))
            dux13=i3(l)*f12(i1(l))*f13(i3(l)-1)*f23(i2(l))
            dux23=i2(l)*f12(i1(l))*f13(i3(l))*f23(i2(l)-1)
         else
            aux1=f12(i1(l))*f13(i3(l))
            aux2=f12(i3(l))*f13(i1(l))
            aux=(aux1+aux2)*f23(i2(l))
            dux23=(aux1+aux2)*i2(l)*f23(i2(l)-1)
            dux1=i1(l)*f12(i1(l)-1)*f13(i3(l))
            dux2=i3(l)*f12(i3(l)-1)*f13(i1(l))
            dux12=(dux1+dux2)*f23(i2(l))
            dux1=i3(l)*f12(i1(l))*f13(i3(l)-1)
            dux2=i1(l)*f12(i3(l))*f13(i1(l)-1)
            dux13=(dux1+dux2)*f23(i2(l))
         endif
         ener=ener+cf(l)*aux
         der12=der12+cf(l)*dux12
         der13=der13+cf(l)*dux13
         der23=der23+cf(l)*dux23
    2 continue
      der(1)=der12*(1.d0-vex1*r12)*dexp(-vex1*r12)
      der(2)=der13*(1.d0-vex1*r13)*dexp(-vex1*r13)
      der(3)=der23*(1.d0-vex2*r23)*dexp(-vex2*r23)
      return
      end
```

Fortran code 6. Analytical expression of the excited 1^3A" PES.

```
      subroutine fit3d(r12,r13,r23,e,der)
      implicit real*8 (a-h,o-z)
      dimension der(3)
      call diat12(r12,e12,d12)
      call diat12(r13,e13,d13)
      call diat23(r23,e23,d23)
      call triabb(r12,r13,r23,e123,der)
      e=e12+e13+e23+e123
      der(1)=d12+der(1)
      der(2)=d13+der(2)
      der(3)=d23+der(3)
      return
      end
*****************************************************************
********
      subroutine diat12(r,ener,der)
*****************************************************************
********
*     This subroutine computes the energies of a diatomic
potential
*     fitted to    54 points
*     rms =      0.05934373 kcal/mol
*     emax =      0.18579253 kcal/mol
*****************************************************************
********
      implicit real*8 (a-h,o-z)
      dimension cf(  8)
      data cf(  1)/0.1059103575538977D+01/
      data cf(  2)/0.4444171059442829D-01/
      data cf(  3)/-.1785173596330693D+01/
      data cf(  4)/0.1389970934818910D+02/
      data cf(  5)/-.5079171234946122D+02/
      data cf(  6)/0.1003419256661402D+03/
      data cf(  7)/-.1024364263643563D+03/
      data cf(  8)/0.4248015432175057D+02/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.5543939378842004D+00
      vex2=0.6299316154588180D+00
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2,  8
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
```

```
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
*************************************************************
********
      subroutine diat23(r,ener,der)
*************************************************************
********
*     This subroutine computes the energies of a diatomic
potential
*     fitted to    72 points
*     rms =      0.02751660 kcal/mol
*     emax =      0.09657713 kcal/mol
*************************************************************
********
      implicit real*8 (a-h,o-z)
      dimension cf( 13)
      data cf(  1)/0.1025911060320943D+01/
      data cf(  2)/-.1175547229303214D+01/
      data cf(  3)/0.2024485840006341D+02/
      data cf(  4)/-.5301786773678648D+03/
      data cf(  5)/0.9979263846994540D+04/
      data cf(  6)/-.1253260333380152D+06/
      data cf(  7)/0.1061566466628005D+07/
      data cf(  8)/-.6141958907006282D+07/
      data cf(  9)/0.2428846391170872D+08/
      data cf( 10)/-.6452782338342221D+08/
      data cf( 11)/0.1100718875325924D+09/
      data cf( 12)/-.1088701923322061D+09/
      data cf( 13)/0.4744897801219261D+08/
      e0=0.0000000000000000D+00
      der=0.d0
      vex1=0.1035656359561501D+01
      vex2=0.1645518387572454D+01
      aux = 1.d0/r
      bux = dexp(-vex2*r)*aux
      cux = dexp(-vex1*r)
      ener=e0+cf(1)*bux
      dux=1.d0
      eux=r*cux
      do 1 i=2, 13
         der=der+(i-1)*cf(i)*dux
         dux=dux*eux
         ener=ener+cf(i)*dux
    1 continue
      der=der*(1.d0-vex1*r)*cux
      der=der-cf(1)*(vex2+aux)*bux
      return
      end
*************************************************************
      subroutine triabb(r12,r13,r23,ener,der)
*************************************************************
*     This subroutine computes the energies of a 3D PES
*     for the ABB system class fitted to **** points
*     rms =      0.95185471 kcal/mol
*     emax =     14.53662118 kcal/mol
```

```
************************************************************
      implicit real*8(a-h,o-z)
      dimension i1(   55),i2(   55),i3(   55),i4(   55),cf(
55)
      dimension f12(0: 6),f13(0: 6),f23(0: 6)
      dimension der(3)
      data cf(  1)/-.3117602534941859D+01/
      data i1(  1)/ 1/,i2(  1)/ 1/,i3(  1)/ 0/,i4(  1)/ 2/
      data cf(  2)/-.6475486874455906D+02/
      data i1(  2)/ 1/,i2(  2)/ 0/,i3(  2)/ 1/,i4(  2)/ 1/
      data cf(  3)/0.1313237768899171D+03/
      data i1(  3)/ 1/,i2(  3)/ 1/,i3(  3)/ 1/,i4(  3)/ 1/
      data cf(  4)/0.3433871915109034D+02/
      data i1(  4)/ 2/,i2(  4)/ 1/,i3(  4)/ 0/,i4(  4)/ 2/
      data cf(  5)/0.3704860445265228D+03/
      data i1(  5)/ 2/,i2(  5)/ 0/,i3(  5)/ 1/,i4(  5)/ 2/
      data cf(  6)/0.2343385052793412D+01/
      data i1(  6)/ 0/,i2(  6)/ 2/,i3(  6)/ 1/,i4(  6)/ 2/
      data cf(  7)/-.6660729001318131D+03/
      data i1(  7)/ 2/,i2(  7)/ 1/,i3(  7)/ 1/,i4(  7)/ 2/
      data cf(  8)/-.1083696590547858D+03/
      data i1(  8)/ 1/,i2(  8)/ 2/,i3(  8)/ 1/,i4(  8)/ 1/
      data cf(  9)/-.2243174000178677D+02/
      data i1(  9)/ 2/,i2(  9)/ 2/,i3(  9)/ 0/,i4(  9)/ 2/
      data cf( 10)/0.2740987234787260D+04/
      data i1( 10)/ 2/,i2( 10)/ 0/,i3( 10)/ 2/,i4( 10)/ 1/
      data cf( 11)/-.1490161829633056D+03/
      data i1( 11)/ 3/,i2( 11)/ 1/,i3( 11)/ 0/,i4( 11)/ 2/
      data cf( 12)/-.4201941535126671D+04/
      data i1( 12)/ 3/,i2( 12)/ 0/,i3( 12)/ 1/,i4( 12)/ 2/
      data cf( 13)/-.7687233351230540D+00/
      data i1( 13)/ 0/,i2( 13)/ 3/,i3( 13)/ 1/,i4( 13)/ 2/
      data cf( 14)/0.4931137377007766D+03/
      data i1( 14)/ 2/,i2( 14)/ 2/,i3( 14)/ 1/,i4( 14)/ 2/
      data cf( 15)/0.3643737382892776D+04/
      data i1( 15)/ 2/,i2( 15)/ 1/,i3( 15)/ 2/,i4( 15)/ 1/
      data cf( 16)/0.3744176347284027D+03/
      data i1( 16)/ 3/,i2( 16)/ 1/,i3( 16)/ 1/,i4( 16)/ 2/
      data cf( 17)/0.3652526884335006D+02/
      data i1( 17)/ 1/,i2( 17)/ 3/,i3( 17)/ 1/,i4( 17)/ 1/
      data cf( 18)/0.6997516359200217D+02/
      data i1( 18)/ 3/,i2( 18)/ 2/,i3( 18)/ 0/,i4( 18)/ 2/
      data cf( 19)/-.1633870892821139D+05/
      data i1( 19)/ 3/,i2( 19)/ 0/,i3( 19)/ 2/,i4( 19)/ 2/
      data cf( 20)/0.5794240507428085D+01/
      data i1( 20)/ 0/,i2( 20)/ 3/,i3( 20)/ 2/,i4( 20)/ 2/
      data cf( 21)/0.3370651151525071D+03/
      data i1( 21)/ 4/,i2( 21)/ 1/,i3( 21)/ 0/,i4( 21)/ 2/
      data cf( 22)/0.3478794751935005D+05/
      data i1( 22)/ 4/,i2( 22)/ 0/,i3( 22)/ 1/,i4( 22)/ 2/
      data cf( 23)/0.1166865850191706D+00/
      data i1( 23)/ 0/,i2( 23)/ 4/,i3( 23)/ 1/,i4( 23)/ 2/
      data cf( 24)/-.2049868365253393D+04/
      data i1( 24)/ 2/,i2( 24)/ 2/,i3( 24)/ 2/,i4( 24)/ 1/
      data cf( 25)/-.4452416147876128D+03/
      data i1( 25)/ 3/,i2( 25)/ 2/,i3( 25)/ 1/,i4( 25)/ 2/
```

```
data cf( 26)/-.4764101748820569D+04/
data i1( 26)/ 3/,i2( 26)/ 1/,i3( 26)/ 2/,i4( 26)/ 2/
data cf( 27)/-.1150955923453323D+03/
data i1( 27)/ 1/,i2( 27)/ 3/,i3( 27)/ 2/,i4( 27)/ 2/
data cf( 28)/-.1077076033175925D+02/
data i1( 28)/ 3/,i2( 28)/ 3/,i3( 28)/ 0/,i4( 28)/ 2/
data cf( 29)/0.9252327303094108D+04/
data i1( 29)/ 3/,i2( 29)/ 0/,i3( 29)/ 3/,i4( 29)/ 1/
data cf( 30)/0.3014464452352719D+04/
data i1( 30)/ 4/,i2( 30)/ 1/,i3( 30)/ 1/,i4( 30)/ 2/
data cf( 31)/-.5136658730245420D+01/
data i1( 31)/ 1/,i2( 31)/ 4/,i3( 31)/ 1/,i4( 31)/ 1/
data cf( 32)/-.1044327970535752D+03/
data i1( 32)/ 4/,i2( 32)/ 2/,i3( 32)/ 0/,i4( 32)/ 2/
data cf( 33)/0.6238506648739926D+05/
data i1( 33)/ 4/,i2( 33)/ 0/,i3( 33)/ 2/,i4( 33)/ 2/
data cf( 34)/-.6646849212097356D+00/
data i1( 34)/ 0/,i2( 34)/ 4/,i3( 34)/ 2/,i4( 34)/ 2/
data cf( 35)/-.3438400332369756D+03/
data i1( 35)/ 5/,i2( 35)/ 1/,i3( 35)/ 0/,i4( 35)/ 2/
data cf( 36)/-.1377096626347440D+06/
data i1( 36)/ 5/,i2( 36)/ 0/,i3( 36)/ 1/,i4( 36)/ 2/
data cf( 37)/-.7683073375143107D-02/
data i1( 37)/ 0/,i2( 37)/ 5/,i3( 37)/ 1/,i4( 37)/ 2/
data cf( 38)/0.1865951776693740D+04/
data i1( 38)/ 3/,i2( 38)/ 2/,i3( 38)/ 2/,i4( 38)/ 2/
data cf( 39)/0.2297528530755202D+03/
data i1( 39)/ 2/,i2( 39)/ 3/,i3( 39)/ 2/,i4( 39)/ 1/
data cf( 40)/0.9879940894068504D+02/
data i1( 40)/ 3/,i2( 40)/ 3/,i3( 40)/ 1/,i4( 40)/ 2/
data cf( 41)/0.1815290127581186D+05/
data i1( 41)/ 3/,i2( 41)/ 1/,i3( 41)/ 3/,i4( 41)/ 1/
data cf( 42)/-.7985212461835086D+03/
data i1( 42)/ 4/,i2( 42)/ 2/,i3( 42)/ 1/,i4( 42)/ 2/
data cf( 43)/-.9221399931516932D+04/
data i1( 43)/ 4/,i2( 43)/ 1/,i3( 43)/ 2/,i4( 43)/ 2/
data cf( 44)/0.7126666406461260D+01/
data i1( 44)/ 1/,i2( 44)/ 4/,i3( 44)/ 2/,i4( 44)/ 2/
data cf( 45)/0.1074825523882955D+02/
data i1( 45)/ 4/,i2( 45)/ 3/,i3( 45)/ 0/,i4( 45)/ 2/
data cf( 46)/-.2827120730400022D+05/
data i1( 46)/ 4/,i2( 46)/ 0/,i3( 46)/ 3/,i4( 46)/ 2/
data cf( 47)/0.4551484476535795D+00/
data i1( 47)/ 0/,i2( 47)/ 4/,i3( 47)/ 3/,i4( 47)/ 2/
data cf( 48)/0.2632956318891673D+03/
data i1( 48)/ 5/,i2( 48)/ 1/,i3( 48)/ 1/,i4( 48)/ 2/
data cf( 49)/0.2742462548440320D+00/
data i1( 49)/ 1/,i2( 49)/ 5/,i3( 49)/ 1/,i4( 49)/ 1/
data cf( 50)/0.7996874138863559D+01/
data i1( 50)/ 5/,i2( 50)/ 2/,i3( 50)/ 0/,i4( 50)/ 2/
data cf( 51)/-.5478780058918209D+05/
data i1( 51)/ 5/,i2( 51)/ 0/,i3( 51)/ 2/,i4( 51)/ 2/
data cf( 52)/0.2957519645413219D-01/
data i1( 52)/ 0/,i2( 52)/ 5/,i3( 52)/ 2/,i4( 52)/ 2/
data cf( 53)/0.4002986033461786D+03/
data i1( 53)/ 6/,i2( 53)/ 1/,i3( 53)/ 0/,i4( 53)/ 2/
```

```
      data cf( 54)/0.1848322458531920D+06/
      data i1( 54)/ 6/,i2( 54)/ 0/,i3( 54)/ 1/,i4( 54)/ 2/
      data cf( 55)/0.1448007484531274D-03/
      data i1( 55)/ 0/,i2( 55)/ 6/,i3( 55)/ 1/,i4( 55)/ 2/
      vex1=0.1402660971663401D+01
      vex2=0.5158151343666982D-01
      f12(0)=1.d0
      f13(0)=1.d0
      f23(0)=1.d0
      bux12=r12*dexp(-vex1*r12)
      bux13=r13*dexp(-vex1*r13)
      bux23=r23*dexp(-vex2*r23)
      do 1 i=1, 6
         f12(i)=f12(i-1)*bux12
         f13(i)=f13(i-1)*bux13
         f23(i)=f23(i-1)*bux23
1     continue
      ener = 0.d0
      der12 = 0.d0
      der13 = 0.d0
      der23 = 0.d0
      do 2 l=1, 55
         if (i4(l).eq.1) then
            aux=f12(i1(l))*f13(i3(l))*f23(i2(l))
            dux12=i1(l)*f12(i1(l)-1)*f13(i3(l))*f23(i2(l))
            dux13=i3(l)*f12(i1(l))*f13(i3(l)-1)*f23(i2(l))
            dux23=i2(l)*f12(i1(l))*f13(i3(l))*f23(i2(l)-1)
         else
            aux1=f12(i1(l))*f13(i3(l))
            aux2=f12(i3(l))*f13(i1(l))
            aux=(aux1+aux2)*f23(i2(l))
            dux23=(aux1+aux2)*i2(l)*f23(i2(l)-1)
            dux1=i1(l)*f12(i1(l)-1)*f13(i3(l))
            dux2=i3(l)*f12(i3(l)-1)*f13(i1(l))
            dux12=(dux1+dux2)*f23(i2(l))
            dux1=i3(l)*f12(i1(l))*f13(i3(l)-1)
            dux2=i1(l)*f12(i3(l))*f13(i1(l)-1)
            dux13=(dux1+dux2)*f23(i2(l))
         endif
         ener=ener+cf(l)*aux
         der12=der12+cf(l)*dux12
         der13=der13+cf(l)*dux13
         der23=der23+cf(l)*dux23
    2 continue
      der(1)=der12*(1.d0-vex1*r12)*dexp(-vex1*r12)
      der(2)=der13*(1.d0-vex1*r13)*dexp(-vex1*r13)
      der(3)=der23*(1.d0-vex2*r23)*dexp(-vex2*r23)
      return
      end
```