

Quantum Mechanics for Chemistry

J. F. Ogilvie

Simon Fraser University, Centre for Experimental and Constructive Mathematics,
Burnaby, British Columbia Canada

This interactive electronic textbook is designed to provide a comprehensive introduction to quantum mechanics in a context of chemistry. This book comprises three extensive chapters, treating model systems, atoms and molecules in turn, applying symbolic calculations with computer program *Maple*. The contents have been generated or collected and adapted from various sources; I am grateful to G. J. Fee, F. M. Fernandez, M. Horbatsch, H. Huber and J. B. Pedersen, among others, for contributions that are acknowledged at the specific locations. 2021 July 9

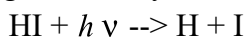
chapter 3 Molecular systems

3.6 diatomic hydrogen as a model for molecular quantum mechanics

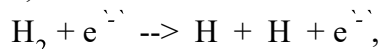
3.61 separation of electronic and nuclear motions

According to either classical mechanics or quantum mechanics, an exact algebraic solution of a system comprising three, or more, particles or bodies is practicable in only a complicated form. A treatment according to theoretical physics of such a system generally relies on approximations in a set specific to the particular problem, and has become among the most computationally intensive fields of science.

Although atomic hydrogen is a well defined chemical species that is observable to participate in significant chemical reactions, after being generated by some process such as photochemical,



or in an electric discharge of molecular hydrogen or other precursor as a result of a collision between that molecule and an energetic electron,



in general the interest of a chemist is devoted to molecular entities and their reactions. The presence and concentration of atomic hydrogen in a particular chemical system might be determined by means of lines in the absorption spectrum, for instance in the Lyman series or specifically at the Lyman α line in the vacuum-ultraviolet region, or the Balmer series in emission. That absorption spectrum is calculated essentially quantitatively in sections 2.53a and 2.54a. Although the practice of atomic absorption for H is hindered because the spectral lines occur in the vacuum-ultraviolet region, the detection and determination of concentration for analytical purposes is practical for many chemical elements with conventional spectrophotometric instruments designed for that purpose of atomic absorption.

A molecule typically comprises multiple atomic nuclei and their multiple associated electrons; for this reason an exact algebraic solution is generally impracticable, and even an accurate approximate solution is challenging of computational resources and duration. A simplification exists when the atomic nuclei are held in fixed relative locations, so with zero kinetic energy, and the associated electrons are left free to move in the field of these stationary nuclei. For the entire system of electrons and nuclei, within the purview of wave mechanics, an assumed total amplitude function $\Psi(r, R)$ depending on the

positions r and momenta p of the electrons and the positions R and momenta P of the nuclei is supposed to be factorable into multiplicand amplitude functions for the electrons, $\psi_e(r)$, and for the nuclei, $\psi_n(R)$, so

$$\Psi(r, R) = \psi_e(r) \psi_n(R)$$

The total hamiltonian operator for the total amplitude function is

$$H_{r, p, R, P} = T_n(P) + T_e(p) + V(r, R)$$

in which appear operators T_n for the kinetic energy of the nuclei of momenta P , T_e for the kinetic energy of the electrons of momenta p , and V for the electrostatic attraction between nuclei and electrons and the corresponding electrostatic repulsions between electrons and between nuclei. According to the treatment of this condition by Born and Oppenheimer in 1927, in the first stage of the calculation, the term for the nuclear kinetic energy T_n is discarded from the total molecular hamiltonian operator, yielding a Schroedinger equation of this form,

$$(T_e + V) \psi_e(r; R) = E(R) \psi_e(r; R)$$

in which the parametric dependence of electronic amplitude function $\psi_e(r; R)$ on the nuclear coordinates is indicated with a symbol after the semicolon; total energy $E(R)$ for the molecular system likewise depends on the relative nuclear coordinates in a particular set. In a prospective second stage of the calculation, that total electronic energy, $E(R)$, becomes a potential energy, V_n , for the motion of the nuclei.

$$(T_n + E(R)) \psi_n(R) = E \psi_n(R)$$

in which the resulting energy E takes values in a discrete or continuous range independent of electronic and nuclear coordinates. The latter equation is hence concerned with the vibrational, rotational and translational motion of the atomic nuclei, which become approximately separable on application of the Eckart conditions. Born and Oppenheimer justified their separate treatment of the nuclear motions only when the surface of potential energy has a unique minimum, local or global, and for only small departures of those pertinent nuclear motions from their equilibrium positions at such a minimum, because the potential energy, $V_n(\xi_i)$, for the motion of the atomic nuclei is expanded in a Taylor series in which ξ_i denote the relative positions of the nuclei with respect to one another. Born subsequently extended that domain (M. Born and K. Huang, *Dynamical Theory of Crystal Lattices*, Oxford University Press, Oxford, UK, 1954) by including some diagonal nonadiabatic effects in the electronic hamiltonian; whereas the Born-Oppenheimer approximation provides a lower bound for the energy of the ground state, the Born-Huang approximation provides an upper bound of that energy.

This separation of electronic and nuclear motions, as an approximation made to facilitate calculations, yields a curve $E(R)$ of total electronic energy for a diatomic molecule as a function of internuclear distance R , or a surface (hypersurface) $E(R_k)$ of total electronic energy for a polyatomic molecule, as a function of all its nuclear coordinates R_k , $1 < k < n$, for the n nuclei in the molecule, in either case theoretically justified for nuclear separations originally corresponding to only the immediate vicinity of a minimum of energy. To yield such a discrete minimum energy, the motion of the centre of mass must be separated, just as was performed for the case of the hydrogen atom; for a stable diatomic molecule, the separate treatment of electronic and nuclear motions then yields a curve, $E(R)$, that has a minimum at a particular internuclear distance or at a set thereof; other discrete curves might be calculated at energies above the minimum energy. The separation becomes practicable because a nuclear mass much exceeds that of the electron, m_e ; the range of validity of this approximation hence

depends on a ratio, to various powers, of electronic and mean nuclear masses, $\frac{m_e}{M_{av}}$; the appropriate

scale of this ratio that Born and Oppenheimer applied is the fourth root, $\kappa = \left(\frac{m_e}{M_{av}} \right)^{\frac{1}{4}}$; the largest value

of the latter quantity for a stable neutral molecule is $\kappa \sim 0.15$ for H_2 . Born and Oppenheimer chose that fourth root of the ratio of masses because, in their perturbation treatment, that power was required to separate the vibrational and rotational terms so that corrections to the energy arise for perturbations of even order, so depending on κ^2 , κ^4 et cetera; as in contemporary mathematical treatments, which involve asymptotic expansions rather than perturbation theory that ultimately diverges, the latter separation is

avoided, the square root suffices, i.e. $\sqrt{\frac{m_e}{M_{av}}} = \kappa^2$. According to the original derivation, the

molecular energy is expanded in a series of which the term of zero order, κ^0 , represents the the energy of the electrons and nuclear repulsion with the nuclei fixed at an optimal configuration; that expansion must be made from a point that represents a minimum of energy or 'equilibrium' conformation. All terms of odd order vanish. The term of second order, κ^2 , represents the energy of vibrations of the nuclei around the optimal configuration in an harmonic approximation; the term of fourth order, κ^4 , contains the rotational energy of the entire molecule, the first anharmonic corrections to the vibrational energy and a first term that couples the nuclear and electronic motions. In many situations, the terms of greater order have no such simple interpretation, but are merely complicated corrections to the expansion of fourth order. In a case of a diatomic molecule of which the molecular state has a non-zero orbital angular momentum about its internuclear axis, terms of further orders introduce splittings of states and their energies that are degenerate at lesser order; for an orbital angular momentum one unit about the internuclear axis, the splitting occurs at eighth order in the expansion, such as for NO in its electronic ground state $X^2\Pi_{\frac{1}{2}, \frac{3}{2}}$. The coefficients of κ to various powers in the expansion are derivatives

$\frac{\partial}{\partial \xi_i} V_n$, $\frac{\partial^2}{\partial \xi_j \partial \xi_i} V_n$, as follows; instead of taking the derivative with respect to ξ directly, ξ is

replaced with $\xi + \kappa \zeta$ and differentiation proceeds with respect to κ . These coefficients of κ^n , being derivatives with respect to ξ_i , are then homogeneous polynomials in ζ_i

$$\begin{aligned} V_n(\xi + \kappa \zeta) &= V_n^{(0)} + \kappa V_n^{(1)} + \kappa^2 V_n^{(2)} + \dots \\ &= V_n(\xi) + \kappa \left(\sum_i \zeta_i \left(\frac{\partial}{\partial \xi_i} V_n \right) \right) + \frac{1}{2} \\ &\quad \kappa^2 \left(\sum_i \sum_j \zeta_i \zeta_j \left(\frac{\partial^2}{\partial \xi_i \partial \xi_j} V_n \right) \right) + \dots \end{aligned}$$

With vanishing terms of odd order, the remaining terms correspond to experimental observations: a vibrational excitation has energy of order κ^2 times an electronic excitation, and a rotational excitation has energy of order κ^4 times an electronic excitation. For a theoretical treatment according to an asymptotic expansion, κ^2 , rather than κ , is hence the parameter most useful to choose.

Born subsequently found this explanation unsatisfactory, although he seemed to retain the notion of a

surface of potential energy with no linear term when expressed in displacement coordinates from a minimum; Born persisted in finding a method to uncouple the electronic states, so retaining the notion of a curve or surface of potential energy, but his explanation has been found to be unsatisfactory.

For an approximate separation of electronic and nuclear motions, the corrections are classified as *adiabatic*, which involve only a single electronic state of interest and are expressed as expectation values of nuclear operators over electronic amplitude functions, and *nonadiabatic*, which connect an electronic state of interest with other electronic states through matrix elements of the same operators between the electronic state of interest and other electronic states of an appropriate symmetry; all these corrections introduce terms into the effective hamiltonian for the vibrational and rotational motions of the atomic nuclei as an adiabatic term to accompany the potential energy, a rotational g factor associated with the centrifugal motion of the nuclei in a state with rotational angular momentum related to matrix elements between the state of interest and other electronic states with angular momentum differing by one unit, and a vibrational g factor associated with the kinetic energy of the nuclei related to matrix elements between the state of interest and other electronic states of the same angular momentum in addition to other nonadiabatic terms of greater order. These g factors might be considered to convey the dependence of the effective, or reduced, mass of the molecule on the internuclear distances. Expectation values of the rotational g factor, g_r , are amenable to separate experimental evaluation in particular electronic and vibrational states through its magnetic effect, but there is no known direct experimental manifestation of the vibrational g factor, g_v . Adiabatic corrections, $U^{BO}(R)$, are purely artefactual, arising from the separation of electronic and nuclear motions and have no direct experimental manifestation.

For an electronic state without contribution to total angular momentum other than from molecular rotation about the centre of mass, the effective Schroedinger equation for nuclear motion of a diatomic molecule of reduced mass μ is hence

$$\left(-\frac{\hbar^2 \left(1 + \frac{m_e g_v(R)}{m_p} \right)}{8 \pi^2 \mu} \frac{\partial^2}{\partial r^2} + U^{BO}(R) + \frac{\left(1 + \frac{m_e g_r(R)}{m_p} \right) \hbar^2 J(J+1)}{8 \pi^2 \mu R^2} + U^{ad}(R) \right) (\psi^{nuc})_{vJM}(R) = E_{vJ} (\psi^{nuc})_{vJM}(R)$$

in which $(\psi^{nuc})_{vJM}(R)$ is the amplitude function for nuclear motion that becomes eventually expressed in terms of quantum numbers vibrational v , rotational J and component M of J ; superscript BO implies an electronic energy calculated at a particular internuclear distance R and ad indicates an adiabatic

correction. In terms of reduced displacement $x = \frac{R - R_e}{R_e}$ and with rotational parameter

$B_e = \frac{h}{8 \pi^2 c \mu R_e^2}$ and energy E_{vJM} in wavenumber units,

$$\left(-B_e \left(1 + \frac{m_e g_v(x)}{m_p} \right) \left(\frac{\partial^2}{\partial x^2} \right) + U_{BO}(x) + \frac{B_e \left(1 + \frac{m_e g_r(x)}{m_p} \right) J(J+1)}{(1+x)^2} + U_{ad}(x) \right) \psi_{vJM}(x) = E_{vJM} \psi_{vJM}(x)$$

In these two equations we have neglected further nonadiabatic terms within the effective hamiltonian on the left sides.

The original article of Born and Oppenheimer is available in English translation at

https://www2.ulb.ac.be/cpm/people/bsutclif/bornopn_corr.pdf

further information about this approximation is available on internet at various locations.

According to Sutcliffe and Woolley [*Journal of Chemical Physics*, 137, 22A544, 2012; 140, 037101, 2014] and contrary to the finding of Born and Oppenheimer of a region, as a segment of a curve or surface, of potential energy near a minimum of total energy, the exact electronic hamiltonian that is expressible as a direct integral of hamiltonians with fixed nuclear coordinates has a purely continuous spectrum of energies above a minimum energy, even after the motion of the molecular centre of mass is separated; those authors concluded that the decomposition of the molecular hamiltonian into a contribution from a nuclear kinetic-energy operator, proportional to κ^4 , and a remainder does not yield a molecular potential-energy surface. This continuum might be a consequence or artefact of that expression as an integral: in either case, there is definable *neither curve nor surface* of potential energy, and hence no basis for the so-called transition-state theory, a basis that was anyhow lacking from the original treatment of Born and Oppenheimer; a constructed curve or surface of potential energy is regarded as a *modification* of quantum mechanics of an empirical nature and *for chemical purposes*. For an hamiltonian of coulombic type to describe one or more atomic nuclei and the associated electrons, if those nuclei be held fixed and if the system be net electrically neutral or positively charged, the electronic hamiltonian has bound states of infinite number at energies less than that energy at a limit corresponding to a dissociation into charged fragments that constitute an ionization; for a molecule, the limit of a dissociation into neutral fragments is not readily defined. Even so, the states of infinite number apply to any geometrical arrangement of the nuclei and for any electronic amplitude function, and consequently to a curve or surface of potential energy anywhere except at nuclear coalescence.

3.62 differential equations for $\text{H}_2^{+'}$

> restart:

Molecular cation $\text{H}_2^{+'}$ is a system comprising two protons and one electron, thus three particles in total. For this reason, we simplify the problem of the dihydrogen molecular cation, $\text{H}_2^{+'}$ by holding two particles, specifically the two atomic nuclei, in fixed relative positions, and then endeavour to calculate the properties of the system as a function of that internuclear distance, according to the first step in the separate treatment of electronic and nuclear motions described in section 3.61. We recognise that this approach is contrary to rigorous quantum mechanics because we thereby fail to treat each particle on the same basis: our approach is equivalent to viewing the electron as subject to quantum mechanics but the two atomic nuclei as subject to classical mechanics, having relative positions subject to no uncertainty.

As a prototypical calculation for a molecule, we hence consider the dihydrogen molecular cation, $\text{H}_2^{+'}$, to comprise two separate and distinct atomic nuclei with labels A and B, of masses m_A and m_B and charges $+Z_A e$ and $+Z_B e$, respectively, and at a fixed distance R from one another, and a single electron of mass μ and charge $-e$ that moves in their electric field, with the distance of the electron from nucleus A being r_A and the distance of the electron from nucleus B being r_B . As in the case of the H atom treated in section 2.53a, we separate the electronic and nuclear motions; in this case there are essentially two nuclear motions, the translation of the entire molecule that has a negligible effect on its internal energies and the translational motion of one atomic nucleus relative to the other and to the electron. The latter motion is, however, significant in that, for a molecule stable with respect to dissociation in its states of least energy, that translational motion of one nucleus relative to the other corresponds to an oscillatory motion, analogous to that considered for the harmonic oscillators in section groups 1.2 and

1.3. When we maintain a fixed distance between those two atomic nuclei, the energy of the system becomes parametrically dependent on that distance, such that the total electronic energy $E(R)$, including the internuclear repulsion, calculated with a fixed internuclear distance becomes a potential energy $V(R)$ for the oscillatory motion of the nuclei when that vibrational degree of freedom is treated subsequently. This separation of electronic and nuclear motions was first considered in detail by Born and Oppenheimer in 1927, although that separation is implicit in preceding discussions of the states of energy of small molecules; such a separation is an approximation, and corrections are required for accurate values of total molecular energies of the discrete states of the molecular system. This approximation is, moreover, valid only in the vicinity of a minimum of electronic energy according to Born and Oppenheimer, despite our capability, demonstrated in what follows, of calculating the energy for internuclear separations far from such a minimum of electronic energy. Although the influence of nuclear mass on the total energy of the system at a fixed internuclear distance is slight, the effect on the energies of the discrete states of the molecular system including the kinetic energy of the atomic nuclei is large. In the succeeding calculations within section group 3.6, we generally ignore these aspects of nuclear mass, and focus on the electronic energy in the electronic state of least energy, hence regarding that energy $E(R)$ to depend parametrically on the internuclear distance.

As also for the hydrogen atom, for the kinetic energy of the electron in the electric field of the two atomic nuclei separate distance R , the laplacian operator thus formed, multiplied by $-\frac{h^2}{8\pi^2\mu}$, is here expressed first in cartesian coordinates.

$$\bar{T} = -\frac{h^2}{8\pi^2\mu} \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right)$$

We suppose the two atomic nuclei A and B to be fixed at locations $(0, 0, z_A) = (0, 0, \frac{R}{2})$ and $(0, 0, z_B) = (0, 0, -\frac{R}{2})$, respectively, along axis z with the origin at the midpoint between them; the electron has general coordinates (x, y, z) . The term for electrostatic potential energy thus becomes

$$V = \frac{e^2}{4\pi\epsilon_0} \left(-\frac{Z_A}{\sqrt{x^2 + y^2 - (z - z_A)^2}} - \frac{Z_B}{\sqrt{x^2 + y^2 + (z - z_B)^2}} + \frac{Z_A Z_B}{R} \right)$$

We make appropriate substitutions in the equation

$$H\psi(x, y, z) = W\psi(x, y, z)$$

to obtain

$$-\frac{h^2}{8\pi^2\mu} \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) \psi(x, y, z) + \frac{e^2}{4\pi\epsilon_0} \left(-\frac{Z_A}{\sqrt{x^2 + y^2 - (z - z_A)^2}} - \frac{Z_B}{\sqrt{x^2 + y^2 + (z - z_B)^2}} + \frac{Z_A Z_B}{R} \right) \psi(x, y, z) = W\psi(x, y, z)$$

which we enter.

```
> Schreq := -h^2/(8*Pi^2*mu)*(diff(psi(x,y,z),x$2)+diff(psi(x,y,z),
y$2)+diff(psi(x,y,z),z$2))
+e^2/(4*Pi*epsilon[0])*(-Z[A]/sqrt(x^2+y^2-(z-z[A])^2)-Z
[B]/sqrt(x^2+y^2+(z-z[B])^2)+Z[A]*Z[B]/R)*psi(x,y,z)
= W*psi(x,y,z);
```

$$\begin{aligned}
Schreq := & - \frac{h^2 \left(\frac{\partial^2}{\partial x^2} \Psi(x, y, z) + \frac{\partial^2}{\partial y^2} \Psi(x, y, z) + \frac{\partial^2}{\partial z^2} \Psi(x, y, z) \right)}{8 \pi^2 \mu} \\
& + \frac{e^2 \left(- \frac{Z_A}{\sqrt{x^2 + y^2 - (z - z_A)^2}} - \frac{Z_B}{\sqrt{x^2 + y^2 + (z - z_B)^2}} + \frac{Z_A Z_B}{R} \right) \Psi(x, y, z)}{4 \pi \epsilon_0} \\
= & W \Psi(x, y, z)
\end{aligned} \tag{2.1}$$

> assume(h>0, Z[A]>0, Z[B]>0, mu>0, W<0, epsilon[0]>0, R>0);

We try to solve this partial-differential equation of second order.

> sol := pdsolve(Schreq, psi(x,y,z));
sol := () (2.2)

As no solution appears, just as in the case of the corresponding partial-differential equation for the H atom in cartesian coordinates, we adopt an alternative strategy. The following derivation and application are based on information in L. C. Pauling and E. B. Wilson, *Introduction to Quantum Mechanics* [McGraw-Hill, New York, USA, 1935] and T. C. Scott, M. Aubert-Frecon and J. Grotendorst, New approach for the electronic energies of the hydrogen molecular ion [*Chemical Physics*, 324, 323--338, 2006], and procedures kindly made available by T. C. Scott.

To take advantage of the prospectively cylindrical symmetry of this molecule comprising two fixed atomic nuclei and one mobile electron, we proceed to work in confocal elliptical, or prolate spheroidal, coordinates, ξ and η , according to these relations that are expressed in terms of distances

$$r_A = \sqrt{x^2 + y^2 - (z - z_A)^2} \quad \text{and} \quad r_B = \sqrt{x^2 + y^2 + (z - z_B)^2}$$

of the electron from the respective atomic nuclei,

$$\begin{aligned}
\xi &= \frac{r_A + r_B}{R} \\
\eta &= \frac{r_A - r_B}{R}
\end{aligned}$$

and azimuthal angle ϕ measured from axis x toward axis y , both perpendicular to the internuclear axis z ; the domain of ξ is $1 \dots \infty$; the domain of η is $-1 \dots 1$; the domain of ϕ is $0 \dots 2\pi$. The surfaces of constant ξ are confocal prolate spheroids (special case of ellipsoids because of symmetry of the two, identical, minor axes) about the line containing atomic nuclei A and B, with these two particles at the foci; the surfaces of constant η are confocal hyperboloids. According to these coordinates, the laplacian operator becomes

$$\frac{4}{R^2 (\xi^2 - \eta^2)} \left(\left(\frac{\partial}{\partial \xi} \right) (\xi^2 - 1) \left(\frac{\partial}{\partial \xi} \right) + \left(\frac{\partial}{\partial \eta} \right) (1 - \eta^2) \left(\frac{\partial}{\partial \eta} \right) + \frac{(\xi^2 - \eta^2) \left(\frac{\partial^2}{\partial \phi^2} \right)}{(\xi^2 - 1) (1 - \eta^2)} \right)$$

> restart:

VectorCalculus:-SetCoordinates('prolatespheroidal'[xi,eta,phi]);
prolatespheroidal _{ξ, η, ϕ} (2.3)

> L := expand(VectorCalculus:-Laplacian(psi(xi,eta,phi)));

$$\begin{aligned}
 L := & \frac{\left(\frac{\partial}{\partial \xi} \psi(\xi, \eta, \phi) \right) \cosh(\xi) \sinh(\xi)}{(-\cos(\eta)^2 + \cosh(\xi)^2) (\cosh(\xi)^2 - 1)} + \frac{\frac{\partial^2}{\partial \xi^2} \psi(\xi, \eta, \phi)}{-\cos(\eta)^2 + \cosh(\xi)^2} \\
 & + \frac{\left(\frac{\partial}{\partial \eta} \psi(\xi, \eta, \phi) \right) \cos(\eta) \sin(\eta)}{(-\cos(\eta)^2 + \cosh(\xi)^2) (1 - \cos(\eta)^2)} + \frac{\frac{\partial^2}{\partial \eta^2} \psi(\xi, \eta, \phi)}{-\cos(\eta)^2 + \cosh(\xi)^2} \\
 & - \frac{\left(\frac{\partial^2}{\partial \phi^2} \psi(\xi, \eta, \phi) \right) \cos(\eta)^2}{(-\cos(\eta)^2 + \cosh(\xi)^2) (1 - \cos(\eta)^2) (\cosh(\xi)^2 - 1)} \\
 & + \frac{\left(\frac{\partial^2}{\partial \phi^2} \psi(\xi, \eta, \phi) \right) \cosh(\xi)^2}{(-\cos(\eta)^2 + \cosh(\xi)^2) (1 - \cos(\eta)^2) (\cosh(\xi)^2 - 1)}
 \end{aligned} \tag{2.4}$$

Schroedinger's temporally independent equation with amplitude function $\psi(\xi, \eta, \phi)$ thus becomes

$$\begin{aligned}
 & - \frac{h^2}{2 \pi^2 R^2 (\xi^2 - \eta^2)} \\
 & \left(\left(\frac{\partial}{\partial \xi} \right) (\xi^2 - 1) \left(\frac{\partial}{\partial \xi} \right) + \left(\frac{\partial}{\partial \eta} \right) (1 - \eta^2) \left(\frac{\partial}{\partial \eta} \right) + \frac{(\xi^2 - \eta^2) \left(\frac{\partial^2}{\partial \phi^2} \right)}{(\xi^2 - 1) (1 - \eta^2)} \right) \psi(\xi, \eta, \phi) + \\
 & \frac{e^2}{4 \pi \epsilon_0} \left(-\frac{2 Z_A}{R (\xi + \eta)} - \frac{2 Z_B}{R (\xi - \eta)} + \frac{Z_A Z_B}{R} \right) \psi(\xi, \eta, \phi) = W \psi(\xi, \eta, \phi),
 \end{aligned}$$

**> Schreq := - h^2/(2*Pi^2*R^2*(xi^2-eta^2))*(diff((xi^2-1)* diff
(psi(xi,eta,phi),xi), xi)
+ diff((1-eta^2)*diff(psi(xi,eta,phi),eta), eta)
+ (xi^2-eta^2)/((xi^2-1)*(1-eta^2))*diff(psi(xi,eta,phi),
phi\$2))
+ e^2/(4*Pi*epsilon[0])*(-2*Z[A]/(R*(xi+eta))-2*Z[B]/
(R*(xi-eta))+Z[A]*Z[B]/R)*psi(xi,eta,phi) = W*psi(xi,eta,phi);**

$$\begin{aligned}
 Schreq := & - \frac{1}{2 \pi^2 R^2 (-\eta^2 + \xi^2)} \left(h^2 \left(2 \xi \left(\frac{\partial}{\partial \xi} \psi(\xi, \eta, \phi) \right) + (\xi^2 - 1) \left(\frac{\partial^2}{\partial \xi^2} \psi(\xi, \eta, \phi) \right) \right. \right. \\
 & \left. \left. - 2 \eta \left(\frac{\partial}{\partial \eta} \psi(\xi, \eta, \phi) \right) + (-\eta^2 + 1) \left(\frac{\partial^2}{\partial \eta^2} \psi(\xi, \eta, \phi) \right) \right) \right. \\
 & \left. + \frac{e^2}{4 \pi \epsilon_0} \left(-\frac{2 Z_A}{R (\xi + \eta)} - \frac{2 Z_B}{R (\xi - \eta)} + \frac{Z_A Z_B}{R} \right) \psi(\xi, \eta, \phi) \right) = W \psi(\xi, \eta, \phi)
 \end{aligned} \tag{2.5}$$

$$+ \frac{(-\eta^2 + \xi^2) \left(\frac{\partial^2}{\partial \phi^2} \psi(\xi, \eta, \phi) \right)}{(\xi^2 - 1)(-\eta^2 + 1)} \Bigg) \\ + \frac{e^2 \left(-\frac{2Z_A}{R(\xi + \eta)} - \frac{2Z_B}{R(\xi - \eta)} + \frac{Z_A Z_B}{R} \right) \psi(\xi, \eta, \phi)}{4\pi\epsilon_0} = W\psi(\xi, \eta, \phi)$$

which we try to solve.

> solSeq := pdsolve(Schreq, psi(xi,eta,phi));

$$solSeq := \psi(\xi, \eta, \phi) = _F3(\phi) _F5(\xi) _F6(\eta) \text{ where } \left[\left\{ \frac{d^2}{d\phi^2} _F3(\phi) = _c3 _F3(\phi), \frac{d^2}{d\eta^2} \right. \right. \quad (2.6)$$

$$_F6(\eta) = \left(-4 _F5(\xi) _F6(\eta) \pi^2 R^2 W \eta^4 \xi^2 \epsilon_0 + 4 _F5(\xi) _F6(\eta) \pi^2 R^2 W \eta^2 \xi^4 \epsilon_0 \right.$$

$$\left. - _F5(\xi) _F6(\eta) \pi R e^2 \eta^4 Z_A Z_B - 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta^3 \xi^2 Z_A \right.$$

$$\left. + 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta^3 \xi^2 Z_B + 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta^2 \xi^3 Z_A \right.$$

$$\left. + 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta^2 \xi^3 Z_B + _F5(\xi) _F6(\eta) \pi R e^2 \xi^4 Z_A Z_B \right.$$

$$\left. - 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta^2 \xi Z_A - 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta^2 \xi Z_B \right.$$

$$\left. + _F5(\xi) _F6(\eta) \pi R e^2 \eta^2 Z_A Z_B + 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta \xi^2 Z_A \right.$$

$$- 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta \xi^2 Z_B - _F5(\xi) _F6(\eta) \pi R e^2 \xi^2 Z_A Z_B + 4 \left(\frac{d}{d\xi} \right.$$

$$\left. _F5(\xi) \right) _F6(\eta) \eta^2 h^2 \xi^3 \epsilon_0 - 4 _F5(\xi) \left(\frac{d}{d\eta} _F6(\eta) \right) \eta^3 h^2 \xi^2 \epsilon_0 - 4 \left(\frac{d}{d\xi} \right.$$

$$\left. _F5(\xi) \right) _F6(\eta) \eta^2 h^2 \xi \epsilon_0 + 4 _F5(\xi) \left(\frac{d}{d\eta} _F6(\eta) \right) \eta h^2 \xi^2 \epsilon_0 - 4 \left(\frac{d}{d\xi} \right.$$

$$\left. _F5(\xi) \right) _F6(\eta) h^2 \xi^3 \epsilon_0 + 4 _F5(\xi) \left(\frac{d}{d\eta} _F6(\eta) \right) \eta^3 h^2 \epsilon_0 + 4 \left(\frac{d}{d\xi} \right.$$

$$\left. _F5(\xi) \right) _F6(\eta) h^2 \xi \epsilon_0 - 4 _F5(\xi) \left(\frac{d}{d\eta} _F6(\eta) \right) \eta h^2 \epsilon_0$$

$$+ 4 _F5(\xi) _F6(\eta) \pi^2 R^2 W \eta^4 \epsilon_0 - 4 _F5(\xi) _F6(\eta) \pi^2 R^2 W \xi^4 \epsilon_0$$

$$- 4 _F5(\xi) _F6(\eta) \pi^2 R^2 W \eta^2 \epsilon_0 + 4 _F5(\xi) _F6(\eta) \pi^2 R^2 W \xi^2 \epsilon_0$$

$$+ 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta^3 Z_A - 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta^3 Z_B$$

$$- 2 _F5(\xi) _F6(\eta) \pi R e^2 \xi^3 Z_A - 2 _F5(\xi) _F6(\eta) \pi R e^2 \xi^3 Z_B$$

$$- 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta Z_A + 2 _F5(\xi) _F6(\eta) \pi R e^2 \eta Z_B$$

$$+ 2 _F5(\xi) _F6(\eta) \pi R e^2 \xi Z_A + 2 _F5(\xi) _F6(\eta) \pi R e^2 \xi Z_B$$

$$- _F5(\xi) _F6(\eta) \pi R e^2 \eta^2 \xi^4 Z_A Z_B + _F5(\xi) _F6(\eta) \pi R e^2 \eta^4 \xi^2 Z_A Z_B$$

$$+ 2 _F5(\xi) _F6(\eta) \eta^2 h^2 _c3 \epsilon_0 - 2 _F5(\xi) _F6(\eta) h^2 \xi^2 _c3 \epsilon_0 - 2 \left(\frac{d^2}{d\xi^2} \right.$$

$$\left. _F5(\xi) \right) _F6(\eta) h^2 \epsilon_0 - 2 \left(\frac{d^2}{d\xi^2} _F5(\xi) \right) _F6(\eta) h^2 \xi^4 \epsilon_0 + 2 \left(\frac{d^2}{d\xi^2} \right.$$

$$\left. _F5(\xi) \right) _F6(\eta) \eta^2 h^2 \epsilon_0 + 4 \left(\frac{d^2}{d\xi^2} _F5(\xi) \right) _F6(\eta) h^2 \xi^2 \epsilon_0 - 4 \left(\frac{d^2}{d\xi^2} \right.$$

$$\left. _F5(\xi) \right) _F6(\eta) \eta^2 h^2 \xi^2 \epsilon_0 + 2 \left(\frac{d^2}{d\xi^2} _F5(\xi) \right) _F6(\eta) \eta^2 h^2 \xi^4 \epsilon_0 \Bigg) /$$

$$\left(2 _F5(\xi) \eta^4 h^2 \xi^2 \epsilon_0 - 4 _F5(\xi) \eta^2 h^2 \xi^2 \epsilon_0 - 2 _F5(\xi) \eta^4 h^2 \epsilon_0 + 2 _F5(\xi) h^2 \xi^2 \epsilon_0 \right.$$

$$\left. + 4 _F5(\xi) \eta^2 h^2 \epsilon_0 - 2 _F5(\xi) h^2 \epsilon_0 \right), \frac{d^3}{d\xi^3} _F5(\xi) = \left(-R e^2 Z_B _F5(\xi)^2 \pi + 4 \left(\frac{d}{d\xi} \right.$$

$$\left. _F5(\xi) \right) h^2 \epsilon_0 _F5(\xi) \xi^2 - R e^2 Z_A _F5(\xi)^2 \pi - 2 \left(\frac{d}{d\xi} _F5(\xi) \right) h^2 \epsilon_0 _F5(\xi) \xi^4$$

$$\begin{aligned}
& -4 R^2 \epsilon_0 W_{F5}(\xi)^2 \pi^2 \xi^5 + 8 R^2 \epsilon_0 W_{F5}(\xi)^2 \pi^2 \xi^3 - R e^2 Z_A F5(\xi)^2 \pi \xi^4 \\
& - R e^2 Z_B F5(\xi)^2 \pi \xi^4 - 4 R^2 \epsilon_0 W_{F5}(\xi)^2 \pi^2 \xi + 2 R e^2 Z_A F5(\xi)^2 \pi \xi^2 \\
& + 2 R e^2 Z_B F5(\xi)^2 \pi \xi^2 + 2 \left(\frac{d}{d\xi} F5(\xi) \right)^2 h^2 \epsilon_0 \xi^5 - 4 \left(\frac{d}{d\xi} F5(\xi) \right)^2 h^2 \epsilon_0 \xi^3 \\
& + 2 \left(\frac{d}{d\xi} F5(\xi) \right)^2 h^2 \epsilon_0 \xi - 2 \left(\frac{d}{d\xi} F5(\xi) \right) h^2 \epsilon_0 F5(\xi) - \left(\frac{d}{d\xi} F5(\xi) \right) \left(\frac{d^2}{d\xi^2} \right. \\
& \left. F5(\xi) \right) h^2 \epsilon_0 + 2 h^2 \epsilon_0 c_3 F5(\xi)^2 \xi + R e^2 Z_A Z_B F5(\xi)^2 \pi \xi^5 - 2 R e^2 Z_A Z_B F5(\xi)^2 \pi \xi^3 \\
& + R e^2 Z_A Z_B F5(\xi)^2 \pi \xi + 3 \left(\frac{d}{d\xi} F5(\xi) \right) \left(\frac{d^2}{d\xi^2} F5(\xi) \right) h^2 \epsilon_0 \xi^2 - 4 \left(\frac{d^2}{d\xi^2} \right. \\
& \left. F5(\xi) \right) h^2 \epsilon_0 F5(\xi) \xi + \left(\frac{d}{d\xi} F5(\xi) \right) \left(\frac{d^2}{d\xi^2} F5(\xi) \right) h^2 \epsilon_0 \xi^6 - 4 \left(\frac{d^2}{d\xi^2} \right. \\
& \left. F5(\xi) \right) h^2 \epsilon_0 F5(\xi) \xi^5 - 3 \left(\frac{d}{d\xi} F5(\xi) \right) \left(\frac{d^2}{d\xi^2} F5(\xi) \right) h^2 \epsilon_0 \xi^4 + 8 \left(\frac{d^2}{d\xi^2} \right. \\
& \left. F5(\xi) \right) h^2 \epsilon_0 F5(\xi) \xi^3 \Bigg) / \left(h^2 \epsilon_0 F5(\xi) \xi^6 - 3 h^2 \epsilon_0 F5(\xi) \xi^4 + 3 F5(\xi) h^2 \xi^2 \epsilon_0 \right. \\
& \left. - F5(\xi) h^2 \epsilon_0 \right) \Bigg]
\end{aligned}$$

A solution of the partial-differential equation appears as the product of three ordinary differential equations, which we separate.

```

> if has(op(1, op(1, op(2, solSeq))), phi) then
    Phieq := op(1, op(1, op(2, solSeq)));
    elif has(op(2, op(1, op(2, solSeq))), phi) then
        Phieq := op(2, op(1, op(2, solSeq)));
    else Phieq := op(3, op(1, op(2, solSeq)));
end if:
Phieq := subs(_F3 = Phi, Phieq);

```

$$Phieq := \frac{d^2}{d\phi^2} \Phi(\phi) = c_3 \Phi(\phi) \quad (2.7)$$

```

> if has(op(1, op(1, op(2, solSeq))), eta) then
    Etaeq := op(1, op(1, op(2, solSeq)));

```



```

elif has(op(2, op(1, op(2, solSeq))), eta) then
  Etaeq := op(2, op(1, op(2, solSeq)));
else Etaeq := op(3, op(1, op(2, solSeq)));
end if:
Etaeq := subs(_F6 = Eta, _F5=Xi, Etaeq);

```

$$Etaeq := \frac{d^2}{d\eta^2} H(\eta) = \left(2 \Xi(\xi) H(\eta) \pi R e^2 \eta^3 \xi^2 Z_B + 2 \Xi(\xi) H(\eta) \pi R e^2 \eta \xi^2 Z_A \right. \quad (2.8)$$

$$+ 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi^3 Z_B - \Xi(\xi) H(\eta) \pi R e^2 \xi^2 Z_A Z_B - 2 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) h^2 \epsilon_0$$

$$- 2 \Xi(\xi) H(\eta) \pi R e^2 \eta \xi^2 Z_B - \Xi(\xi) H(\eta) \pi R e^2 \eta^4 Z_A Z_B + 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi^3 Z_A$$

$$- 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi Z_A + \Xi(\xi) H(\eta) \pi R e^2 \eta^2 Z_A Z_B - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^3 \xi^2 Z_A$$

$$- 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \eta^4 \xi^2 \epsilon_0 - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi Z_B$$

$$+ 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \eta^2 \xi^4 \epsilon_0 + \Xi(\xi) H(\eta) \pi R e^2 \xi^4 Z_A Z_B - 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \eta^2 \epsilon_0$$

$$+ 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \xi^2 \epsilon_0 + 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^3 Z_A - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^3 Z_B$$

$$- 2 \Xi(\xi) H(\eta) \pi R e^2 \xi^3 Z_A - 2 \Xi(\xi) H(\eta) \pi R e^2 \xi^3 Z_B - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta Z_A$$

$$+ 2 \Xi(\xi) H(\eta) \pi R e^2 \eta Z_B + 2 \Xi(\xi) H(\eta) \pi R e^2 \xi Z_A + 2 \Xi(\xi) H(\eta) \pi R e^2 \xi Z_B$$

$$+ 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \eta^4 \epsilon_0 - 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \xi^4 \epsilon_0 + 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) h^2 \xi \epsilon_0$$

$$- 4 \Xi(\xi) \left(\frac{d}{d\eta} H(\eta) \right) \eta h^2 \epsilon_0 - 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) h^2 \xi^3 \epsilon_0 + 4 \Xi(\xi) \left(\frac{d}{d\eta} \right.$$

$$H(\eta) \left. \right) \eta^3 h^2 \epsilon_0 + 2 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \epsilon_0 + 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) h^2 \xi^2 \epsilon_0 - 2 \left(\frac{d^2}{d\xi^2} \right.$$

$$\Xi(\xi) \left. \right) H(\eta) h^2 \xi^4 \epsilon_0 + 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \xi^3 \epsilon_0 - 4 \Xi(\xi) \left(\frac{d}{d\eta} H(\eta) \right) \eta^3 h^2 \xi^2 \epsilon_0$$

$$\begin{aligned}
& -4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \xi \epsilon_0 + 4 \Xi(\xi) \left(\frac{d}{d\eta} H(\eta) \right) \eta h^2 \xi^2 \epsilon_0 \\
& + 2 \Xi(\xi) H(\eta) \eta^2 h^2 \epsilon_0 - 2 \Xi(\xi) H(\eta) h^2 \xi^2 \epsilon_0 - 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \xi^2 \epsilon_0 \\
& + 2 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \xi^4 \epsilon_0 - \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi^4 Z_A Z_B \\
& + \Xi(\xi) H(\eta) \pi R e^2 \eta^4 \xi^2 Z_A Z_B \Bigg) / \left(2 \Xi(\xi) \eta^4 h^2 \xi^2 \epsilon_0 - 4 \Xi(\xi) \eta^2 h^2 \xi^2 \epsilon_0 \right. \\
& \left. - 2 \Xi(\xi) \eta^4 h^2 \epsilon_0 + 2 \Xi(\xi) h^2 \xi^2 \epsilon_0 + 4 \Xi(\xi) \eta^2 h^2 \epsilon_0 - 2 \Xi(\xi) h^2 \epsilon_0 \right)
\end{aligned}$$

```

> if has(lhs(op(1, op(1, op(2, solSeq)))), xi) then
  Xieq := op(1, op(1, op(2, solSeq)));
elif has(lhs(op(2, op(1, op(2, solSeq)))), xi) then
  Xieq := op(2, op(1, op(2, solSeq)));
else Xieq := op(3, op(1, op(2, solSeq)));
end if:
Xieq := subs(_F5 = Xi, _F6 = Eta, Xieq);

```

$$Xieq := \frac{d^3}{d\xi^3} \Xi(\xi) = \left(2 \left(\frac{d}{d\xi} \Xi(\xi) \right)^2 h^2 \epsilon_0 \xi^5 - 4 \left(\frac{d}{d\xi} \Xi(\xi) \right)^2 h^2 \epsilon_0 \xi^3 + 2 \left(\frac{d}{d\xi} \Xi(\xi) \right)^2 h^2 \epsilon_0 \xi - 2 \left(\frac{d}{d\xi} \right. \right. \quad (2.$$

$$\begin{aligned}
& \Xi(\xi) \Big) h^2 \epsilon_0 \Xi(\xi) - \left(\frac{d}{d\xi} \Xi(\xi) \right) \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 + R e^2 Z_A Z_B \Xi(\xi)^2 \pi \xi^5 \\
& - 2 R e^2 Z_A Z_B \Xi(\xi)^2 \pi \xi^3 + R e^2 Z_A Z_B \Xi(\xi)^2 \pi \xi - R e^2 Z_B \Xi(\xi)^2 \pi + 4 \left(\frac{d}{d\xi} \right. \\
& \Xi(\xi) \Big) h^2 \epsilon_0 \Xi(\xi) \xi^2 - R e^2 Z_A \Xi(\xi)^2 \pi - 2 \left(\frac{d}{d\xi} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) \xi^4 + 2 h^2 \epsilon_0 \epsilon_3 \Xi(\xi)^2 \xi \\
& + 3 \left(\frac{d}{d\xi} \Xi(\xi) \right) \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \xi^2 - 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) \xi + \left(\frac{d}{d\xi} \right. \\
& \Xi(\xi) \Big) \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \xi^6 - 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) \xi^5 - 3 \left(\frac{d}{d\xi} \Xi(\xi) \right) \left(\frac{d^2}{d\xi^2} \right. \\
& \Xi(\xi) \Big) h^2 \epsilon_0 \xi^4 + 8 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) \xi^3 + 2 R e^2 Z_A \Xi(\xi)^2 \pi \xi^2 + 2 R e^2 Z_B \Xi(\xi)^2 \pi \xi^2 \\
& - 4 R^2 \epsilon_0 W \Xi(\xi)^2 \pi^2 \xi^5 + 8 R^2 \epsilon_0 W \Xi(\xi)^2 \pi^2 \xi^3 - R e^2 Z_A \Xi(\xi)^2 \pi \xi^4 - R e^2 Z_B \Xi(\xi)^2 \pi \xi^4
\end{aligned}$$

$$\left. -4 R^2 \epsilon_0 W \Xi(\xi)^2 \pi^2 \xi \right) / \left(h^2 \epsilon_0 \Xi(\xi) \xi^6 - 3 h^2 \epsilon_0 \Xi(\xi) \xi^4 + 3 \Xi(\xi) h^2 \xi^2 \epsilon_0 \right. \\ \left. - \Xi(\xi) h^2 \epsilon_0 \right)$$

> Etaeq := subs(_c[3]=-m^2, Etaeq);

$$Etaeq := \frac{d^2}{d\eta^2} H(\eta) = \left(-2 \Xi(\xi) H(\eta) \eta^2 h^2 m^2 \epsilon_0 + 2 \Xi(\xi) H(\eta) h^2 \xi^2 m^2 \epsilon_0 \right. \quad (2.10)$$

$$\left. + 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^3 \xi^2 Z_B + 2 \Xi(\xi) H(\eta) \pi R e^2 \eta \xi^2 Z_A \right. \\ + 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi^3 Z_B - \Xi(\xi) H(\eta) \pi R e^2 \xi^2 Z_A Z_B - 2 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) h^2 \epsilon_0 \\ - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta \xi^2 Z_B - \Xi(\xi) H(\eta) \pi R e^2 \eta^4 Z_A Z_B + 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi^3 Z_A \\ - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi Z_A + \Xi(\xi) H(\eta) \pi R e^2 \eta^2 Z_A Z_B - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^3 \xi^2 Z_A \\ - 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \eta^4 \xi^2 \epsilon_0 - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi Z_B \\ + 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \eta^2 \xi^4 \epsilon_0 + \Xi(\xi) H(\eta) \pi R e^2 \xi^4 Z_A Z_B - 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \eta^2 \epsilon_0 \\ + 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \xi^2 \epsilon_0 + 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^3 Z_A - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta^3 Z_B \\ - 2 \Xi(\xi) H(\eta) \pi R e^2 \xi^3 Z_A - 2 \Xi(\xi) H(\eta) \pi R e^2 \xi^3 Z_B - 2 \Xi(\xi) H(\eta) \pi R e^2 \eta Z_A \\ + 2 \Xi(\xi) H(\eta) \pi R e^2 \eta Z_B + 2 \Xi(\xi) H(\eta) \pi R e^2 \xi Z_A + 2 \Xi(\xi) H(\eta) \pi R e^2 \xi Z_B \\ + 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \eta^4 \epsilon_0 - 4 \Xi(\xi) H(\eta) \pi^2 R^2 W \xi^4 \epsilon_0 + 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) h^2 \xi \epsilon_0 \\ - 4 \Xi(\xi) \left(\frac{d}{d\eta} H(\eta) \right) \eta h^2 \epsilon_0 - 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) h^2 \xi^3 \epsilon_0 + 4 \Xi(\xi) \left(\frac{d}{d\eta} \right. \\ \left. H(\eta) \right) \eta^3 h^2 \epsilon_0 + 2 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \epsilon_0 + 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) h^2 \xi^2 \epsilon_0$$

$$\begin{aligned}
& -2 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) h^2 \xi^4 \epsilon_0 + 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \xi^3 \epsilon_0 - 4 \Xi(\xi) \left(\frac{d}{d\eta} H(\eta) \right) \eta^3 h^2 \xi^2 \epsilon_0 \\
& - 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \xi \epsilon_0 + 4 \Xi(\xi) \left(\frac{d}{d\eta} H(\eta) \right) \eta h^2 \xi^2 \epsilon_0 \\
& - 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \xi^2 \epsilon_0 + 2 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) \eta^2 h^2 \xi^4 \epsilon_0 \\
& - \Xi(\xi) H(\eta) \pi R e^2 \eta^2 \xi^4 Z_A Z_B + \Xi(\xi) H(\eta) \pi R e^2 \eta^4 \xi^2 Z_A Z_B \Bigg) / \left(2 \Xi(\xi) \eta^4 h^2 \xi^2 \epsilon_0 \right. \\
& \left. - 4 \Xi(\xi) \eta^2 h^2 \xi^2 \epsilon_0 - 2 \Xi(\xi) \eta^4 h^2 \epsilon_0 + 2 \Xi(\xi) h^2 \xi^2 \epsilon_0 + 4 \Xi(\xi) \eta^2 h^2 \epsilon_0 - 2 \Xi(\xi) h^2 \epsilon_0 \right)
\end{aligned}$$

> Xieq := subs(_c[3]=-m^2, Xieq);

$$\begin{aligned}
Xieq := & \frac{d^3}{d\xi^3} \Xi(\xi) = \left(-2 h^2 \epsilon_0 m^2 \Xi(\xi)^2 \xi + 2 \left(\frac{d}{d\xi} \Xi(\xi) \right)^2 h^2 \epsilon_0 \xi^5 - 4 \left(\frac{d}{d\xi} \Xi(\xi) \right)^2 h^2 \epsilon_0 \xi^3 \right. \\
& + 2 \left(\frac{d}{d\xi} \Xi(\xi) \right)^2 h^2 \epsilon_0 \xi - 2 \left(\frac{d}{d\xi} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) - \left(\frac{d}{d\xi} \Xi(\xi) \right) \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \\
& + R e^2 Z_A Z_B \Xi(\xi)^2 \pi \xi^5 - 2 R e^2 Z_A Z_B \Xi(\xi)^2 \pi \xi^3 + R e^2 Z_A Z_B \Xi(\xi)^2 \pi \xi - R e^2 Z_B \Xi(\xi)^2 \pi \\
& + 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) \xi^2 - R e^2 Z_A \Xi(\xi)^2 \pi - 2 \left(\frac{d}{d\xi} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) \xi^4 \\
& + 3 \left(\frac{d}{d\xi} \Xi(\xi) \right) \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \xi^2 - 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) \xi + \left(\frac{d}{d\xi} \Xi(\xi) \right) \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \xi^6 \\
& - 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) \xi^5 - 3 \left(\frac{d}{d\xi} \Xi(\xi) \right) \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \xi^4 \\
& + 8 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) h^2 \epsilon_0 \Xi(\xi) \xi^3 + 2 R e^2 Z_A \Xi(\xi)^2 \pi \xi^2 \\
& + 2 R e^2 Z_B \Xi(\xi)^2 \pi \xi^2 - 4 R^2 \epsilon_0 W \Xi(\xi)^2 \pi^2 \xi^5 + 8 R^2 \epsilon_0 W \Xi(\xi)^2 \pi^2 \xi^3 - R e^2 Z_A \Xi(\xi)^2 \pi \xi^4 \\
& \left. - R e^2 Z_B \Xi(\xi)^2 \pi \xi^4 - 4 R^2 \epsilon_0 W \Xi(\xi)^2 \pi^2 \xi \right) / \left(h^2 \epsilon_0 \Xi(\xi) \xi^6 - 3 h^2 \epsilon_0 \Xi(\xi) \xi^4 \right. \\
& \left. + 3 \Xi(\xi) h^2 \xi^2 \epsilon_0 - \Xi(\xi) h^2 \epsilon_0 \right)
\end{aligned} \tag{2.11}$$

As the equation for $H(\eta)$ contains also $\Xi(\xi)$, we try to obtain a solution for both functions in a system.

```
> sol := dsolve({Xieq,Etaeq}, {Xi(xi),Eta(eta)});
```

$$sol := \left\{ \begin{array}{l} H(\eta) = {}_2C4 \left((\eta + 1) (\eta - 1) \right)^{\frac{m}{2}} \end{array} \right. \quad (2.12)$$

$$+ \frac{1}{2} e^{\frac{\sqrt{\pi} (\eta + 1)}{2 h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} - 2 \left(\int \frac{\eta}{\eta^2 - 1} d\eta \right) h} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right. \right.$$

$$\left. \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \right), m, m, \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0},$$

$$\left. \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R \left((Z_B + 2) Z_A - 2 Z_B \right) + \epsilon_0 h^2 (m^2 + 2 {}_2C1)}{2 \epsilon_0 h^2}, \frac{\eta}{2} + \frac{1}{2} \right)$$

$$+ {}_2C5 e^{\frac{\sqrt{\pi} (\eta + 1)}{2 h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} - 2 \left(\int \frac{\eta}{\eta^2 - 1} d\eta \right) h} \left(\frac{\eta}{2} + \frac{1}{2} \right)^{-\frac{m}{2}}$$

$$+ \frac{1}{2} \left(\frac{\eta}{2} - \frac{1}{2} \right)^{\frac{m}{2} + \frac{1}{2}} \operatorname{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}{h}, -m, m,$$

$$\frac{2 \pi R e^2 \left(Z_A - Z_B \right)}{h^2 \epsilon_0}, \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R \left(\left(Z_B + 2 \right) Z_A - 2 Z_B \right) + \epsilon_0 h^2 \left(m^2 + 2_{-C1} \right)}{2 \epsilon_0 h^2},$$

$$\left. \frac{\eta}{2} + \frac{1}{2} \right), \Xi(\xi)$$

$$=_{-C2} e^{\frac{\sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \xi}{2 h}} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right.$$

$$\left. \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \right), m, m, \frac{2 \pi R e^2 \left(Z_A + Z_B \right)}{h^2 \epsilon_0},$$

$$\left. \frac{4 \pi^2 R^2 W \epsilon_0 - \pi \left(\left(Z_B + 2 \right) Z_A + 2 Z_B \right) e^2 R + \epsilon_0 h^2 \left(m^2 + 2_{-C1} \right)}{2 \epsilon_0 h^2}, \frac{\xi}{2} + \frac{1}{2} \right) \left(\xi^2 - 1 \right)^{\frac{m}{2}}$$

$$+_{-C3} e^{\frac{\sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \xi}{2 h}} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right.$$

$$\left. \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \right), -m, m, \frac{2 \pi R e^2 \left(Z_A + Z_B \right)}{h^2 \epsilon_0},$$

$$\left. \frac{4\pi^2 R^2 W \epsilon_0 - \pi \left((Z_B + 2) Z_A + 2 Z_B \right) e^2 R + \epsilon_0 h^2 (m^2 + 2 _C1)}{2 \epsilon_0 h^2}, \frac{\xi}{2} + \frac{1}{2} \right\} \\ \sqrt{2\xi - 2} (\xi + 1)^{-\frac{m}{2}} (\xi - 1)^{\frac{m}{2} - \frac{1}{2}} \left. \right\}$$

If in some release of *Maple* the preceding command yields no output, we can apply the result from another release or version.

```
> sola := {Eta(eta) = _C4*((eta-1)*(eta+1))^(1/2*m+1/2)*exp(1/2*(Pi^(1/2)*(eta+1)*(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B]))*R/epsilon[0])^(1/2)-2*Int(eta/(eta^2-1),eta)*h)/h)*HeunC(2/h*Pi^(1/2)*(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B]))*R/epsilon[0])^(1/2),m,m,2/h^2/epsilon[0]*Pi*R*e^2*(-Z[B]+Z[A]),1/2*(4*Pi^2*R^2*W*epsilon[0]-R*((Z[B]+2)*Z[A]-2*Z[B]))*e^2*Pi+epsilon[0]*h^2*(m^2+2*_C1))/h^2/epsilon[0],1/2*eta+1/2)+_C5*exp(1/2*(Pi^(1/2)*(eta+1)*(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B]))*R/epsilon[0])^(1/2)-2*Int(eta/(eta^2-1),eta)*h)/h)*(1/2*eta+1/2)^(-1/2*m+1/2)*(1/2*eta-1/2)^(1/2*m+1/2)*HeunC(2/h*Pi^(1/2)*(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B]))*R/epsilon[0])^(1/2),-m,m,2/h^2/epsilon[0]*Pi*R*e^2*(-Z[B]+Z[A]),1/2*(4*Pi^2*R^2*W*epsilon[0]-R*((Z[B]+2)*Z[A]-2*Z[B]))*e^2*Pi+epsilon[0]*h^2*(m^2+2*_C1))/h^2/epsilon[0],1/2*eta+1/2), Xi(xi) = _C2*exp(1/2*1/h*Pi^(1/2)*(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B]))*R/epsilon[0])^(1/2)*xi)*HeunC(2/h*Pi^(1/2)*(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B]))*R/epsilon[0])^(1/2),m,m,2*Pi*R*e^2*(Z[A]+Z[B])/h^2/epsilon[0],1/2*(4*Pi^2*R^2*W*epsilon[0]-((Z[B]+2)*Z[A]+2*Z[B]))*R*e^2*Pi+epsilon[0]*h^2*(m^2+2*_C1))/epsilon[0]/h^2,1/2*xi+1/2)*(xi^2-1)^(1/2*m)+_C3*exp(1/2*1/h*Pi^(1/2)*(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B]))*R/epsilon[0])^(1/2)*xi)*HeunC(2/h*Pi^(1/2)*(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B]))*R/epsilon[0])^(1/2),-m,m,2*Pi*R*e^2*(Z[A]+Z[B])/h^2/epsilon[0],1/2*(4*Pi^2*R^2*W*epsilon[0]-((Z[B]+2)*Z[A]+2*Z[B]))*R*e^2*Pi+epsilon[0]*h^2*(m^2+2*_C1))/epsilon[0]/h^2,1/2*xi+1/2)*(2*xi-2)^(1/2)*(xi+1)^(-1/2*m)*(xi-1)^(1/2*m-1/2)}:
```

We separate the solutions for $\Xi(\xi)$ and $H(\eta)$.

```
> Xisol := op(2, sol);
```

$Xisol := \Xi(\xi)$

(2.13)

$$\begin{aligned}
&= {}_{-C2} e^{\frac{\sqrt{\pi}}{2h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \xi} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right. \right. \\
&\quad \left. \left. \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \right), m, m, \frac{2 \pi R e^2 (Z_A + Z_B)}{h^2 \epsilon_0}, \right. \\
&\quad \left. \frac{4 \pi^2 R^2 W \epsilon_0 - \pi \left((Z_B + 2) Z_A + 2 Z_B \right) e^2 R + \epsilon_0 h^2 (m^2 + 2 {}_{-C1})}{2 \epsilon_0 h^2}, \frac{\xi}{2} + \frac{1}{2} \right) (\xi^2 - 1)^{\frac{m}{2}} \\
&+ {}_{-C3} e^{\frac{\sqrt{\pi}}{2h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \xi} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right. \right. \\
&\quad \left. \left. \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \right), -m, m, \frac{2 \pi R e^2 (Z_A + Z_B)}{h^2 \epsilon_0}, \right. \\
&\quad \left. \frac{4 \pi^2 R^2 W \epsilon_0 - \pi \left((Z_B + 2) Z_A + 2 Z_B \right) e^2 R + \epsilon_0 h^2 (m^2 + 2 {}_{-C1})}{2 \epsilon_0 h^2}, \frac{\xi}{2} + \frac{1}{2} \right) \\
&\quad \sqrt{2 \xi - 2} (\xi + 1)^{-\frac{m}{2}} (\xi - 1)^{\frac{m}{2} - \frac{1}{2}}
\end{aligned}$$

> Etasol := op(1, sol);

$$Etasol := H(\eta) = {}_{-C4} ((\eta + 1) (\eta - 1))^{\frac{m}{2}} \quad (2.14)$$

$$+ \frac{1}{2} e^{\frac{\sqrt{\pi} (\eta + 1)}{2h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} - 2 \left(\left[\frac{\eta}{\eta^2 - 1} d\eta \right] h} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right. \right.$$

$$\begin{aligned}
& \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \Bigg), m, m, \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \\
& \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R \left((Z_B + 2) Z_A - 2 Z_B \right) + \epsilon_0 h^2 (m^2 + 2 _Cl)}{2 \epsilon_0 h^2}, \frac{\eta}{2} + \frac{1}{2} \Bigg) \\
& + _C5 e^{\frac{\sqrt{\pi} (\eta + 1)}{2 h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} - 2 \left(\int \frac{\eta}{\eta^2 - 1} d\eta \right) h} \left(\frac{\eta}{2} + \frac{1}{2} \right)^{-\frac{m}{2}} \\
& + \frac{1}{2} \left(\frac{\eta}{2} - \frac{1}{2} \right)^{\frac{m}{2} + \frac{1}{2}} \text{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}{h}, -m, m, \right. \\
& \left. \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R \left((Z_B + 2) Z_A - 2 Z_B \right) + \epsilon_0 h^2 (m^2 + 2 _Cl)}{2 \epsilon_0 h^2}, \right. \\
& \left. \frac{\eta}{2} + \frac{1}{2} \right)
\end{aligned}$$

```

> Eta(eta) := _C4*((eta-1)*(eta+1))^(1/2*m+1/2)*exp(1/2*(Pi^(1/2)*
(eta+1)*(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B])*R/epsilon[0])^
(1/2)/h))*exp(-2*Int(eta/(eta^2-1),eta))*HeunC(2/h*Pi^(1/2)*(-8*
(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B])*R/epsilon[0])^(1/2),m,m,
2/h^2/epsilon[0]*Pi*R*e^2*(-Z[B]+Z[A]),1/2*(4*Pi^2*R^2*W*epsilon
[0]-R*((Z[B]+2)*Z[A]-2*Z[B])*e^2*Pi+epsilon[0]*h^2*(m^2+2*_C1))
/h^2/epsilon[0],1/2*eta+1/2)+_C5*exp(1/2*(Pi^(1/2)*(eta+1)*(-8*
(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B])*R/epsilon[0])^(1/2)/h))*exp
(-2*Int(eta/(eta^2-1),eta))*(1/2*eta+1/2)^(-1/2*m+1/2)*(1/2*
eta-1/2)^(1/2*m+1/2)*HeunC(2/h*Pi^(1/2)*(-8*(Pi*R*W*epsilon[0]

```

$$-1/4*e^2*Z[A]*Z[B])*R/epsilon[0])^(1/2),-m,m,2/h^2/epsilon[0]*Pi*\\R*e^2*(-Z[B]+Z[A]),1/2*(4*Pi^2*R^2*W*epsilon[0]-R*((Z[B]+2)*Z[A]-2*Z[B])*e^2*Pi+epsilon[0]*h^2*(m^2+2*_C1))/h^2/epsilon[0],1/2*eta+1/2);$$

$$H(\eta) := {}_4C4 \left((\eta + 1) (\eta - 1) \right)^{\frac{m}{2}} \quad (2.15)$$

$$+ \frac{1}{2} \frac{\sqrt{\pi} (\eta + 1)}{e} \frac{\sqrt{\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}{2 h} e^{-2 \left(\int \frac{\eta}{\eta^2 - 1} d\eta \right)} \text{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right. \right. \\ \left. \left. \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \right), m, m, \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \right. \\ \left. \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2 {}_C1)}{2 \epsilon_0 h^2}, \frac{\eta}{2} + \frac{1}{2} \right) \\ + {}_4C5 e \frac{\sqrt{\pi} (\eta + 1)}{e} \frac{\sqrt{\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}{2 h} e^{-2 \left(\int \frac{\eta}{\eta^2 - 1} d\eta \right)} \left(\frac{\eta}{2} + \frac{1}{2} \right)^{-\frac{m}{2}} \\ + \frac{1}{2} \left(\frac{\eta}{2} - \frac{1}{2} \right)^{\frac{m}{2} + \frac{1}{2}} \text{HeunC} \left(\frac{2 \sqrt{\pi}}{h} \frac{\sqrt{\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}{h}, -m, m, \right. \\ \left. \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2 {}_C1)}{2 \epsilon_0 h^2} \right),$$

$$\left. \frac{\eta}{2} + \frac{1}{2} \right)$$

The solutions for $H(\eta)$ contain an integral that we evaluate, and insert the result into the solutions.

> simplify(exp(-2*int(eta/(eta^2-1),eta)));

$$\frac{1}{\eta^2 - 1} \quad (2.16)$$

> Eta(eta) := subs(exp(-2*Int(eta/(eta^2-1),eta))=1/(eta^2-1), Eta(eta));

$$\begin{aligned} H(\eta) := & \frac{1}{\eta^2 - 1} \left(-C4 \left((\eta + 1) (\eta - 1) \right)^{\frac{m}{2}} \right. \\ & + \frac{1}{2} e^{\frac{\sqrt{\pi} (\eta + 1)}{2h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right. \right. \\ & \left. \left. \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}, m, m, \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \right. \right. \\ & \left. \left. \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R \left((Z_B + 2) Z_A - 2 Z_B \right) + \epsilon_0 h^2 (m^2 + 2_{-CI})}{2 \epsilon_0 h^2}, \frac{\eta}{2} + \frac{1}{2} \right) \right) \\ & \left. + \frac{1}{\eta^2 - 1} \left(-C5 e^{\frac{\sqrt{\pi} (\eta + 1)}{2h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}} \left(\frac{\eta}{2} + \frac{1}{2} \right)^{-\frac{m}{2} + \frac{1}{2}} \left(\frac{\eta}{2} \right. \right. \right. \end{aligned} \quad (2.17)$$

$$-\frac{1}{2}\left)^{\frac{m}{2}+\frac{1}{2}}\text{HeunC}\left(\frac{2\sqrt{\pi}\sqrt{-\frac{8\left(\pi R W\epsilon_0-\frac{e^2 Z_A Z_B}{4}\right)R}{\epsilon_0}}}{h},-m,m,\right. \\ \left.\frac{2\pi R e^2\left(Z_A-Z_B\right)}{h^2\epsilon_0},\frac{4\pi^2 R^2 W\epsilon_0-\pi e^2 R\left(\left(Z_B+2\right)Z_A-2Z_B\right)+\epsilon_0 h^2\left(m^2+2_Cl\right)}{2\epsilon_0 h^2},\right. \\ \left.\left.\frac{\eta}{2}+\frac{1}{2}\right)\right)$$

We separate the two parts of each solution for $\Xi(\xi)$ and $H(\eta)$.

> Xisol1 := subs(_C3=0, Xisol);

$Xisol1 := \Xi(\xi)$ (2.18)

$$=_C2 e^{\frac{\sqrt{\pi}}{2h}\sqrt{-\frac{8\left(\pi R W\epsilon_0-\frac{e^2 Z_A Z_B}{4}\right)R}{\epsilon_0}}\xi}\text{HeunC}\left(\frac{1}{h}\left(2\sqrt{\pi}\sqrt{-\frac{8\left(\pi R W\epsilon_0-\frac{e^2 Z_A Z_B}{4}\right)R}{\epsilon_0}}\right),m,m,\frac{2\pi R e^2\left(Z_A+Z_B\right)}{h^2\epsilon_0},\right. \\ \left.\frac{4\pi^2 R^2 W\epsilon_0-\pi\left(\left(Z_B+2\right)Z_A+2Z_B\right)e^2 R+\epsilon_0 h^2\left(m^2+2_Cl\right)}{2\epsilon_0 h^2},\frac{\xi}{2}+\frac{1}{2}\right)\left(\xi^2-1\right)^{\frac{m}{2}}$$

> Xisol1 := Xi(xi) = _C2*exp(1/2*1/h*Pi^(1/2)*(expand(-8*R*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B])/epsilon[0]))^(1/2)*xi)*HeunC(2/h*Pi^(1/2)*(-8*R*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B])/epsilon[0]))^(1/2),m,m,2*Pi*R*e^2*(Z[A]+Z[B])/h^2/epsilon[0],expand(1/2*(4*Pi^2*R^2*W*epsilon[0]-e^2*R*Pi*((Z[B]+2)*Z[A]+2*Z[B])+h^2*epsilon[0]*(m^2+2*_Cl))/h^2/epsilon[0]),1/2*xi+1/2)*(xi^2-1)

$$\text{Xisol1} := \Xi(\xi) \quad (2.19)$$

$$= {}_{-C2} e^{\frac{\sqrt{\pi}}{2h} \sqrt{-8\pi R^2 W + \frac{2R e^2 Z_A Z_B}{\epsilon_0}} \xi} \text{HeunC} \left(\frac{2\sqrt{\pi}}{h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}, m, m, \frac{2\pi R e^2 (Z_A + Z_B)}{h^2 \epsilon_0}, \frac{2\pi^2 R^2 W}{h^2} - \frac{\pi R e^2 Z_A Z_B}{2h^2 \epsilon_0} - \frac{\pi R e^2 Z_A}{h^2 \epsilon_0} - \frac{\pi R e^2 Z_B}{h^2 \epsilon_0} + \frac{m^2}{2} + {}_{-C1}, \frac{\xi}{2} + \frac{1}{2} \right) (\xi^2 - 1)^{\frac{m}{2}}$$

$$\text{Xisol2} := \Xi(\xi) \quad (2.20)$$

$$= {}_{-C3} e^{\frac{\sqrt{\pi}}{2h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \xi} \text{HeunC} \left(\frac{1}{h} \sqrt{2\sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}, -m, m, \frac{2\pi R e^2 (Z_A + Z_B)}{h^2 \epsilon_0}, \frac{4\pi^2 R^2 W \epsilon_0 - \pi \left((Z_B + 2) Z_A + 2 Z_B \right) e^2 R + \epsilon_0 h^2 (m^2 + 2 {}_{-C1})}{2 \epsilon_0 h^2}, \frac{\xi}{2} + \frac{1}{2} \right) \sqrt{2\xi - 2} (\xi + 1)^{-\frac{m}{2}} (\xi - 1)^{\frac{m}{2} - \frac{1}{2}}$$

$$\begin{aligned} &> \text{Xisol2} := \text{Xi}(\text{xi}) = {}_{-C3} \sqrt{2} \exp(1/2 * 1/h * \text{Pi}^{(1/2)} * (\text{expand}(-8 * R * \\ &(\text{Pi} * R * W * \text{epsilon}[0] - 1/4 * e^2 * Z[A] * Z[B]) / \text{epsilon}[0]))^{(1/2)} * \text{xi}) * \text{HeunC} \\ &(2/h * \text{Pi}^{(1/2)} * (\text{expand}(-8 * R * (\text{Pi} * R * W * \text{epsilon}[0] - 1/4 * e^2 * Z[A] * Z[B]) \\ &/ \text{epsilon}[0]))^{(1/2)}, -m, m, 2 * \text{Pi} * R * e^2 * (Z[A] + Z[B]) / h^2 / \text{epsilon}[0], \\ &\text{expand}(1/2 * (4 * \text{Pi}^2 * R^2 * W * \text{epsilon}[0] - \text{Pi} * R * ((Z[B] + 2) * Z[A] + 2 * Z[B]) * \\ &e^2 + h^2 * \text{epsilon}[0] * (m^2 + 2 * {}_{-C1})) / h^2 / \text{epsilon}[0]), 1/2 * \text{xi} + 1/2) * (\\ &(\text{xi} - 1) / (\text{xi} + 1))^{(m/2)}; \end{aligned}$$

$$Xisol2 := \Xi(\xi) \quad (2.21)$$

$$= {}_3C_3 \sqrt{2} e^{\frac{\sqrt{\pi} \sqrt{-8\pi R^2 W + \frac{2 R e^2 Z_A Z_B}{\epsilon_0}} \xi}{2h}} \text{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-8\pi R^2 W + \frac{2 R e^2 Z_A Z_B}{\epsilon_0}}}{h}, \right. \\ \left. -m, m, \frac{2\pi R e^2 (Z_A + Z_B)}{h^2 \epsilon_0}, \frac{2\pi^2 R^2 W}{h^2} - \frac{\pi R e^2 Z_A Z_B}{2h^2 \epsilon_0} - \frac{\pi R e^2 Z_A}{h^2 \epsilon_0} - \frac{\pi R e^2 Z_B}{h^2 \epsilon_0} + \frac{m^2}{2} \right. \\ \left. + {}_3C_1, \frac{\xi}{2} + \frac{1}{2} \right) \left(\frac{\xi - 1}{\xi + 1} \right)^{\frac{m}{2}}$$

$$> \text{Etasol1} := \text{subs}(_C5=0, \text{Eta}(\eta));$$

$$\text{Etasol1} := \frac{1}{\eta^2 - 1} \left(-C4 ((\eta + 1)(\eta - 1))^{\frac{m}{2}} \right. \\ \left. + \frac{1}{2} e^{\frac{\sqrt{\pi} (\eta + 1) \sqrt{-8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}}{\epsilon_0}}}{2h} \text{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \sqrt{-8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R} \right. \right. \right. \\ \left. \left. \left. - \frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0} \right), m, m, \frac{2\pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \right. \right. \\ \left. \left. \frac{4\pi^2 R^2 W \epsilon_0 - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2 {}_3C_1)}{2 \epsilon_0 h^2}, \frac{\eta}{2} + \frac{1}{2} \right) \right) \quad (2.22)$$

$$> \text{Etasol1} := \text{subs}((\eta-1)=(1-\eta^2)/(\eta+1), \text{Etasol1});$$

$$(2.23)$$

$$\begin{aligned}
Etasol1 := & \frac{1}{\eta^2 - 1} \left(-C4 \left(-\eta^2 + 1 \right)^{\frac{m}{2}} \right. \\
& + \frac{1}{2} \frac{\sqrt{\pi} (\eta + 1)}{e} \frac{\sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}{2h} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right. \right. \\
& \left. \left. \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \right), m, m, \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \right. \\
& \left. \left. \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2 _Cl)}{2 \epsilon_0 h^2}, \frac{\eta}{2} + \frac{1}{2} \right) \right)
\end{aligned} \tag{2.23}$$

```

> Etasol1 := _C4*(-eta^2+1)^(1/2*m-1/2)*exp(1/2*Pi^(1/2)*(eta+1)*
(expand(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B]))*R/epsilon[0]))^
(1/2)/h)*HeunC(2/h*Pi^(1/2)*(expand(-8*(Pi*R*W*epsilon[0]-1/4*e^2*
Z[A]*Z[B]))*R/epsilon[0]))^(1/2),m,m,2/h^2/epsilon[0]*Pi*R*e^2*(-Z
[B]+Z[A]),expand(1/2*(4*Pi^2*R^2*W*epsilon[0]-R*((Z[B]+2)*Z[A]-2*Z
[B])*e^2*Pi+epsilon[0]*h^2*(m^2+2*_Cl))/h^2/epsilon[0]),1/2*
eta+1/2);

```

$$\begin{aligned}
Etasol1 := & _C4 \left(-\eta^2 + 1 \right)^{\frac{m}{2}} \\
& - \frac{1}{2} \frac{\sqrt{\pi} (\eta + 1)}{e} \frac{\sqrt{-8 \pi R^2 W + \frac{2 R e^2 Z_A Z_B}{\epsilon_0}}}{2h} \operatorname{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-8 \pi R^2 W + \frac{2 R e^2 Z_A Z_B}{\epsilon_0}}}{h}, \right. \\
& m, m, \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \frac{2 \pi^2 R^2 W}{h^2} - \frac{\pi R e^2 Z_A Z_B}{2 h^2 \epsilon_0} - \frac{\pi R e^2 Z_A}{h^2 \epsilon_0} + \frac{\pi R e^2 Z_B}{h^2 \epsilon_0} + \frac{m^2}{2} \\
& \left. + _Cl, \frac{\eta}{2} + \frac{1}{2} \right)
\end{aligned} \tag{2.24}$$

> Etasol2 := subs(_C4=0, Eta(eta));

$$Etasol2 := \frac{1}{\eta^2 - 1} \left(-C5 e^{\frac{\sqrt{\pi} (\eta + 1) \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}}{\epsilon_0}}{2h}} \left(\frac{\eta}{2} + \frac{1}{2} \right)^{-\frac{m}{2} + \frac{1}{2}} \left(\frac{\eta}{2} \right. \right. \quad (2.25)$$

$$\left. \left. - \frac{1}{2} \right)^{\frac{m}{2} + \frac{1}{2}} \text{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}}{\epsilon_0}}{h}, -m, m, \right. \right.$$

$$\left. \left. \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2 _C1)}{2 \epsilon_0 h^2}, \right. \right.$$

$$\left. \left. \frac{\eta}{2} + \frac{1}{2} \right) \right)$$

> Etasol2 := 1/2* _C5*exp(1/2*Pi^(1/2)*(eta+1)*(expand(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B])*R/epsilon[0]))^(1/2)/h)*(1-eta^2)^(1/2*m-1/2)*HeunC(2/h*Pi^(1/2)*(expand(-8*(Pi*R*W*epsilon[0]-1/4*e^2*Z[A]*Z[B])*R/epsilon[0]))^(1/2),-m,m,2/h^2/epsilon[0]*Pi*R*e^2*(-Z[B]+Z[A]),expand(1/2*(4*Pi^2*R^2*W*epsilon[0]-R*((Z[B]+2)*Z[A]-2*Z[B])*e^2*Pi+epsilon[0]*h^2*(m^2+2*_C1))/h^2/epsilon[0]),1/2*eta+1/2);

$$Etasol2 := \frac{1}{2} \left(-C5 e^{\frac{\sqrt{\pi} (\eta + 1) \sqrt{-8 \pi R^2 W + \frac{2 R e^2 Z_A Z_B}{\epsilon_0}}}{2h}} (-\eta^2 + 1)^{\frac{m}{2}} \right. \quad (2.26)$$

$$\left. - \frac{1}{2} \text{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-8 \pi R^2 W + \frac{2 R e^2 Z_A Z_B}{\epsilon_0}}}{h}, -m, m, \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \right. \right.$$

$$\left. \frac{2 \pi^2 R^2 W}{h^2} - \frac{\pi R e^2 Z_A Z_B}{2 h^2 \epsilon_0} - \frac{\pi R e^2 Z_A}{h^2 \epsilon_0} + \frac{\pi R e^2 Z_B}{h^2 \epsilon_0} + \frac{m^2}{2} + {}_C I, \frac{\eta}{2} + \frac{1}{2} \right) \right)$$

We solve the separate ordinary-differential equation for $\Phi(\phi)$.

> Phisol := dsolve(Phieq, Phi(phi));

$$Phisol := \Phi(\phi) = {}_C I e^{\sqrt{-c_3} \phi} + {}_C 2 e^{-\sqrt{-c_3} \phi} \quad (2.27)$$

The solution for the $\Phi(\phi)$ equation is the same as for the H atom; we recall that solution here in normalized form.

> Phisol := Phi(phi) = 1/2*1/Pi^(1/2)*2^(1/2)*exp(m*phi*I);

$$Phisol := \Phi(\phi) = \frac{\sqrt{2} e^{Im \phi}}{2 \sqrt{\pi}} \quad (2.28)$$

In the above solutions for $\Xi(\xi)$ and $H(\eta)$, ${}_C 1$ has the role of a parameter that enters through the separation of the variables. Although in principle we have derived the amplitude functions for this system, in practice these functions are intractable in this form.

For a special case of interest for ${}^1H_2^{+'}$, $Z_A = Z_B = 1$ and $m = 0$,

> Xisol1 := subs(Z[A]=Z[B], Z[B]=1, m=0, {}_C 2=N, Xisol1);

$$Xisol1 := \Xi(\xi) = N e^{\frac{\sqrt{\pi} \sqrt{-8 \pi R^2 W + \frac{2 R e^2}{\epsilon_0}} \xi}{2 h}} \text{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2}{4} \right) R}{\epsilon_0}}}{h}, \quad (2.29)$$

$$0, 0, \frac{4 \pi R e^2}{h^2 \epsilon_0}, \frac{2 \pi^2 R^2 W}{h^2} - \frac{5 \pi R e^2}{2 h^2 \epsilon_0} + {}_C I, \frac{\xi}{2} + \frac{1}{2} \right)$$

> Xisol2 := subs(Z[A]=Z[B], Z[B]=1, m=0, {}_C 3=N/sqrt(2), Xisol2);

$$Xisol2 := \Xi(\xi) = N e^{\frac{\sqrt{\pi} \sqrt{-8 \pi R^2 W + \frac{2 R e^2}{\epsilon_0}} \xi}{2 h}} \text{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-8 \pi R^2 W + \frac{2 R e^2}{\epsilon_0}}}{h}, 0, 0, \quad (2.30)$$

$$\frac{4 \pi R e^2}{h^2 \epsilon_0}, \frac{2 \pi^2 R^2 W}{h^2} - \frac{5 \pi R e^2}{2 h^2 \epsilon_0} + {}_C I, \frac{\xi}{2} + \frac{1}{2} \right)$$

the two solutions for $\Xi(\xi)$ become identical, with normalising factor N in each case.

> Etasol1 := Eta(eta) = subs(Z[A]=Z[B], Z[B]=1, m=0, _C4=N, Etasol1)
;

$$\begin{aligned}
 Etasol1 := & \frac{1}{\eta^2 - 1} \left(-C4 \left((\eta + 1) (\eta - 1) \right)^{\frac{m}{2}} \right. \\
 & + \frac{1}{2} \frac{\sqrt{\pi} (\eta + 1)}{e^{\frac{\sqrt{\pi} (\eta + 1)}{2h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right. \right. \\
 & \left. \left. \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}} \right), m, m, \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \right. \\
 & \left. \left. \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2 _C1)}{2 \epsilon_0 h^2}, \frac{\eta}{2} + \frac{1}{2} \right) \right) \\
 & + \frac{1}{\eta^2 - 1} \left(-C5 e^{\frac{\sqrt{\pi} (\eta + 1)}{2h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}} \left(\frac{\eta}{2} + \frac{1}{2} \right)^{-\frac{m}{2} + \frac{1}{2}} \left(\frac{\eta}{2} \right. \right. \\
 & \left. \left. - \frac{1}{2} \right)^{\frac{m}{2} + \frac{1}{2}} \operatorname{HeunC} \left(\frac{2 \sqrt{\pi}}{h} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}, -m, m, \right. \right. \\
 & \left. \left. \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2 _C1)}{2 \epsilon_0 h^2} \right), \right.
 \end{aligned} \tag{2.31}$$

$$\left. \left. \left. \left. \left. \frac{\eta}{2} + \frac{1}{2} \right) \right) \right) \right) \right)$$

$$= \frac{1}{\sqrt{-\eta^2 + 1}} \left(N e^{\frac{\sqrt{\pi} (\eta + 1) \sqrt{-8 \pi R^2 W + \frac{2 R e^2}{\epsilon_0}}}{2 h}} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \sqrt{-8 \pi R^2 W + \frac{2 R e^2}{\epsilon_0}} \right), 0, 0, 0, \frac{2 \pi^2 R^2 W}{h^2} - \frac{\pi R e^2}{2 h^2 \epsilon_0} + _C I, \frac{\eta}{2} + \frac{1}{2} \right) \right)$$

> Etasol2 := Eta(eta) = subs(Z[A]=Z[B], Z[B]=1, m=0, _C5=2*N, (eta/2+1/2)^(1/2)=(eta^2-1)^(1/2)/2/(eta/2-1/2)^(1/2), Etasol2);

$$Etasol2 := \frac{1}{\eta^2 - 1} \left(-C4 ((\eta + 1) (\eta - 1))^{\frac{m}{2}} \right. \tag{2.32}$$

$$+ \frac{1}{2} e^{\frac{\sqrt{\pi} (\eta + 1) \sqrt{-8 \left(\pi R W \epsilon_0 - \frac{\mathcal{C}^2 Z_A Z_B}{4} \right) R}}{2 h}} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \sqrt{-8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R} \right), m, m, \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \right.$$

$$\left. \left. \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2 _C I)}{2 \epsilon_0 h^2}, \frac{\eta}{2} + \frac{1}{2} \right) \right)$$

$$\begin{aligned}
& + \frac{1}{\eta^2 - 1} \left(-C5 e^{\frac{\sqrt{\pi}(\eta+1)}{2h} \sqrt{-\frac{8\left(\pi R W_{\epsilon_0} - \frac{e^2 Z_A Z_B}{4}\right)R}{\epsilon_0}}} \left(\frac{\eta}{2} + \frac{1}{2}\right)^{-\frac{m}{2} + \frac{1}{2}} \left(\frac{\eta}{2} - \frac{1}{2}\right)^{\frac{m}{2} + \frac{1}{2}} \text{HeunC}\left(\frac{2\sqrt{\pi} \sqrt{-\frac{8\left(\pi R W_{\epsilon_0} - \frac{e^2 Z_A Z_B}{4}\right)R}{\epsilon_0}}}{h}, -m, m, \right. \right. \\
& \left. \left. \frac{2\pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \frac{4\pi^2 R^2 W_{\epsilon_0} - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2 - CI)}{2 \epsilon_0 h^2}, \right. \right. \\
& \left. \left. \frac{\eta}{2} + \frac{1}{2} \right) \right) \\
& = \frac{1}{\sqrt{-\eta^2 + 1}} \left(N e^{\frac{\sqrt{\pi}(\eta+1)}{2h} \sqrt{-8\pi R^2 W + \frac{2R e^2}{\epsilon_0}}} \text{HeunC}\left(\frac{1}{h} \left(2\sqrt{\pi} \sqrt{-8\pi R^2 W + \frac{2R e^2}{\epsilon_0}}\right), 0, 0, 0, \frac{2\pi^2 R^2 W}{h^2} - \frac{\pi R e^2}{2 h^2 \epsilon_0} + -CI, \frac{\eta}{2} + \frac{1}{2} \right) \right)
\end{aligned}$$

With the appropriate substitutions, the two solutions for $H(\eta)$ become identical. To this point and under the special condition for $^1H_2^{+}$, the amplitude function is simply a product of $\Xi(\xi)$ and $H(\eta)$ because $\Phi(\phi) = 0$ for $m = 0$.

> psi(xi,eta) := rhs(Xisoll)*Etasoll/N;

$$\psi(\xi, \eta) := e^{\frac{\sqrt{\pi} \sqrt{-8\pi R^2 W + \frac{2R e^2}{\epsilon_0}}}{2h} \xi} \text{HeunC}\left(\frac{2\sqrt{\pi} \sqrt{-\frac{8\left(\pi R W_{\epsilon_0} - \frac{e^2}{4}\right)R}{\epsilon_0}}}{h}, 0, 0, 0, \frac{2\pi^2 R^2 W}{h^2} - \frac{\pi R e^2}{2 h^2 \epsilon_0} + -CI, \frac{\eta}{2} + \frac{1}{2} \right) \quad (2.33)$$

$$\frac{4 \pi R e^2}{h^2 \epsilon_0}, \frac{2 \pi^2 R^2 W}{h^2} - \frac{5 \pi R e^2}{2 h^2 \epsilon_0} + {}_2C1, \frac{\xi}{2} + \frac{1}{2} \Bigg) \Bigg(\frac{1}{\eta^2 - 1} \Bigg(- {}_4C4 \left((\eta + 1) \left(\eta \right. \right.$$

$$\left. - 1 \right) \right)^{\frac{m}{2}}$$

$$+ \frac{1}{2} \frac{\sqrt{\pi} \left(\eta + 1 \right) \sqrt{- \frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}{2 h} \operatorname{HeunC} \left(\frac{1}{h} \left(2 \sqrt{\pi} \right. \right.$$

$$\left. \left. - \frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0} \right) \right), m, m, \frac{2 \pi R e^2 \left(Z_A - Z_B \right)}{h^2 \epsilon_0},$$

$$\frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R \left(\left(Z_B + 2 \right) Z_A - 2 Z_B \right) + \epsilon_0 h^2 \left(m^2 + 2 {}_2C1 \right)}{2 \epsilon_0 h^2}, \frac{\eta}{2} + \frac{1}{2} \Bigg) \Bigg)$$

$$+ \frac{1}{\eta^2 - 1} \Bigg(- {}_5C5 \frac{\sqrt{\pi} \left(\eta + 1 \right) \sqrt{- \frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}{\epsilon_0}}}{2 h} \left(\frac{\eta}{2} + \frac{1}{2} \right)^{- \frac{m}{2} + \frac{1}{2}} \left(\frac{\eta}{2} \right.$$

$$\begin{aligned}
& -\frac{1}{2} \Big)^{\frac{m}{2} + \frac{1}{2}} \text{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2 Z_A Z_B}{4} \right) R}}{\epsilon_0}}{h}, -m, m, \right. \\
& \left. \frac{2 \pi R e^2 (Z_A - Z_B)}{h^2 \epsilon_0}, \frac{4 \pi^2 R^2 W \epsilon_0 - \pi e^2 R ((Z_B + 2) Z_A - 2 Z_B) + \epsilon_0 h^2 (m^2 + 2_{-CI})}{2 \epsilon_0 h^2}, \right. \\
& \left. \left. \left. \left. \frac{\eta}{2} + \frac{1}{2} \right) \right) \right) \right) \\
& = \frac{1}{\sqrt{-\eta^2 + 1}} \left(e^{\frac{\sqrt{\pi} \sqrt{-8 \pi R^2 W + \frac{2 R e^2}{\epsilon_0}} \xi}{2 h}} \text{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2}{4} \right) R}}{\epsilon_0}}{h}, \right. \right. \\
& \left. \left. 0, 0, \frac{4 \pi R e^2}{h^2 \epsilon_0}, \frac{2 \pi^2 R^2 W}{h^2} - \frac{5 \pi R e^2}{2 h^2 \epsilon_0} +_{-CI}, \frac{\xi}{2} + \frac{1}{2} \right) \right. \\
& \left. N e^{\frac{\sqrt{\pi} (\eta + 1) \sqrt{-8 \pi R^2 W + \frac{2 R e^2}{\epsilon_0}}}{2 h}} \text{HeunC} \left(\frac{2 \sqrt{\pi} \sqrt{-8 \pi R^2 W + \frac{2 R e^2}{\epsilon_0}}}{h}, 0, 0, 0, \right. \right. \\
& \left. \left. \frac{2 \pi^2 R^2 W}{h^2} - \frac{\pi R e^2}{2 h^2 \epsilon_0} +_{-CI}, \frac{\eta}{2} + \frac{1}{2} \right) \right)
\end{aligned}$$

With some manipulation to make the solution more compact, we obtain this expression.

```

> psi(xi,eta) := N*exp(-1/2*1/h*Pi^(1/2)*(-8*Pi*R^2*W+2/epsilon[0]*
R*e^2)^(1/2)*(xi+eta+1))*HeunC(2/h*Pi^(1/2)*(-8*R*(Pi*R*W*epsilon

```

$$[0]-1/4*e^2)/\epsilon_0)^{(1/2)},0,0,4/\epsilon_0/h^2*\pi*R*e^2, \\ 2/h^2*\pi^2*R^2*W+_C1-5/2*1/\epsilon_0/h^2*\pi*R*e^2,1/2*\xi+1/2)/(- \\ \eta^2+1)^{(1/2)}*HeunC(2*\pi^{(1/2)}*(-8*\pi*R^2*W+2/\epsilon_0*R*e^2)^{ \\ (1/2)}/h,0,0,0,2/h^2*\pi^2*R^2*W+_C1-1/2*1/\epsilon_0/h^2*\pi*R*e^2, \\ 1/2*\eta+1/2);$$

$$\psi(\xi, \eta) :=$$

(2.34)

$$\frac{1}{\sqrt{-\eta^2+1}} \left(N e^{-\frac{\sqrt{\pi} \sqrt{-8\pi R^2 W + \frac{2R e^2}{\epsilon_0}} (\xi + \eta + 1)}{2h}} \text{HeunC} \left(\frac{1}{h} \left(2\sqrt{\pi} \sqrt{-\frac{8 \left(\pi R W \epsilon_0 - \frac{e^2}{4} \right) R}{\epsilon_0}}, 0, 0, \frac{4\pi R e^2}{h^2 \epsilon_0}, \frac{2\pi^2 R^2 W}{h^2} - \frac{5\pi R e^2}{2h^2 \epsilon_0} + _C1, \frac{\xi}{2} + \frac{1}{2} \right) \right. \right. \\ \left. \left. \text{HeunC} \left(\frac{2\sqrt{\pi} \sqrt{-8\pi R^2 W + \frac{2R e^2}{\epsilon_0}}}{h}, 0, 0, 0, \frac{2\pi^2 R^2 W}{h^2} - \frac{\pi R e^2}{2h^2 \epsilon_0} + _C1, \frac{\eta}{2} + \frac{1}{2} \right) \right) \right)$$

If a plot of this amplitude function were practicable for a particular value of W , it would be cylindrically symmetric about axis z that contains the two protons as atomic nuclei. This simplified amplitude function has three parameters -- internuclear separation R , energy $W(R)$ that is a function of R and separation parameter $_C1(R)$ that is also a function of R . At a particular internuclear separation R , the task is to find consonant values of W and $_C1$ so as to define not only amplitude function $\psi(\xi, \eta)$ but also the function for potential energy for each electronic state. For a bound state, solution of another Schroedinger equation for the nuclear motion with $W(R)$ as a function for potential energy -- i.e. the electronic total energy as a function of R becomes a potential energy for the nuclear motion -- yields the energies of the vibrational states in that electronic state of molecular ion $^1\text{H}_2^{+'}$. This problem is too complicated for an exact analytic (algebraic) solution at present.

We try to simplify the problem, following Pauling and Wilson, by working in atomic units in which $m_e = \frac{h}{2\pi} = e = 1$ and with $Z_A = Z_B = 1$, so that the molecular cation becomes precisely $^1\text{H}_2^{+'}$; the amplitude equation becomes

$$\frac{\partial}{\partial \xi} \left((\xi^2 - 1) \left(\frac{\partial}{\partial \xi} \psi \right) \right) + \frac{\partial}{\partial \eta} \left((1 - \eta^2) \left(\frac{\partial}{\partial \eta} \psi \right) \right) + \left(\frac{1}{\xi^2 - 1} + \frac{1}{1 - \eta^2} \right) \left(\frac{\partial^2}{\partial \phi^2} \psi \right) \\ + 2R^2 \left(\frac{W'(\xi^2 - \eta^2)}{4} + \frac{\xi}{R} \right) \psi = 0$$

in which electronic energy $W' = W - \frac{1}{R}$, the relation $\frac{1}{r_A} + \frac{1}{r_B} = \frac{4\xi}{R(\xi^2 - \eta^2)}$ has been used, and we have multiplied through by $\frac{R(\xi^2 - \eta^2)}{4}$.

> restart:

VectorCalculus:-SetCoordinates('prolatespheroidal'[xi,eta,phi]);
prolatespheroidal _{ξ, η, ϕ} (2.35)

> Schreq := Diff((xi^2-1)* Diff(psi(xi,eta,phi),xi), xi)
+ Diff((1-eta^2)*Diff(psi(xi,eta,phi), eta), eta)
+ (1/(xi^2-1)+1/(1-eta^2))*Diff(psi(xi,eta,phi),phi\$2)
+ 2*R^2*(`W'`*(xi^2-eta^2)/4+xi/R)*psi(xi,eta,phi) = 0;

$$\begin{aligned} Schreq := & \frac{\partial}{\partial \xi} \left((\xi^2 - 1) \left(\frac{\partial}{\partial \xi} \psi(\xi, \eta, \phi) \right) \right) + \frac{\partial}{\partial \eta} \left((-\eta^2 + 1) \left(\frac{\partial}{\partial \eta} \psi(\xi, \eta, \phi) \right) \right) \\ & + \left(\frac{1}{\xi^2 - 1} + \frac{1}{-\eta^2 + 1} \right) \left(\frac{\partial^2}{\partial \phi^2} \psi(\xi, \eta, \phi) \right) + 2R^2 \left(\frac{W'(-\eta^2 + \xi^2)}{4} + \frac{\xi}{R} \right) \psi(\xi, \eta, \phi) = 0 \end{aligned} \quad (2.36)$$

We solve this partial-differential equation,

> solSeq := pdsolve(Schreq, psi(xi,eta,phi));

$$solSeq := \psi(\xi, \eta, \phi) = {}_F3(\phi) {}_F5(\xi) {}_F6(\eta) \text{ where } \left\{ \frac{d^2}{d\phi^2} {}_F3(\phi) = {}_c3 {}_F3(\phi), \frac{d^2}{d\eta^2} \right. \quad (2.37)$$

$$\begin{aligned} {}_F6(\eta) = & \left(4 {}_F5(\xi) \left(\frac{d}{d\eta} {}_F6(\eta) \right) \eta^3 - 4 \left(\frac{d}{d\xi} {}_F5(\xi) \right) {}_F6(\eta) \xi^3 \right. \\ & - 4 {}_F5(\xi) \left(\frac{d}{d\eta} {}_F6(\eta) \right) \eta + 4 \left(\frac{d}{d\xi} {}_F5(\xi) \right) {}_F6(\eta) \xi + {}_F5(\xi) {}_F6(\eta) R^2 W' \eta^4 \\ & - {}_F5(\xi) {}_F6(\eta) R^2 W' \xi^4 + 4 {}_F5(\xi) {}_F6(\eta) R \eta^2 \xi^3 - {}_F5(\xi) {}_F6(\eta) R^2 W' \eta^2 \\ & \left. + {}_F5(\xi) {}_F6(\eta) R^2 W' \xi^2 - 4 {}_F5(\xi) {}_F6(\eta) R \eta^2 \xi + 4 {}_F5(\xi) \left(\frac{d}{d\eta} {}_F6(\eta) \right) \eta \xi^2 \right) \end{aligned}$$

$$\begin{aligned}
& -4 \left(\frac{d}{d\xi} {}_F5(\xi) \right) {}_F6(\eta) \eta^2 \xi + 4 {}_F5(\xi) {}_F6(\eta) R \xi - 4 {}_F5(\xi) \left(\frac{d}{d\eta} \right. \\
& \left. {}_F6(\eta) \right) \eta^3 \xi^2 + 4 \left(\frac{d}{d\xi} {}_F5(\xi) \right) {}_F6(\eta) \eta^2 \xi^3 - 4 {}_F5(\xi) {}_F6(\eta) R \xi^3 \\
& - {}_F5(\xi) {}_F6(\eta) R^2 W' \eta^4 \xi^2 + {}_F5(\xi) {}_F6(\eta) R^2 W' \eta^2 \xi^4 - 2 \left(\frac{d^2}{d\xi^2} \right. \\
& \left. {}_F5(\xi) \right) {}_F6(\eta) \xi^4 + 2 \left(\frac{d^2}{d\xi^2} {}_F5(\xi) \right) {}_F6(\eta) \eta^2 + 4 \left(\frac{d^2}{d\xi^2} {}_F5(\xi) \right) {}_F6(\eta) \xi^2 \\
& - 2 \left(\frac{d^2}{d\xi^2} {}_F5(\xi) \right) {}_F6(\eta) - 4 \left(\frac{d^2}{d\xi^2} {}_F5(\xi) \right) {}_F6(\eta) \eta^2 \xi^2 + 2 \left(\frac{d^2}{d\xi^2} \right. \\
& \left. {}_F5(\xi) \right) {}_F6(\eta) \eta^2 \xi^4 + 2 {}_F5(\xi) {}_F6(\eta) \eta^2 {}_{c_3} - 2 {}_F5(\xi) {}_F6(\eta) \xi^2 {}_{c_3} \Big) \Big/ \\
& (2 {}_F5(\xi) \eta^4 \xi^2 - 4 {}_F5(\xi) \eta^2 \xi^2 - 2 {}_F5(\xi) \eta^4 + 2 {}_F5(\xi) \xi^2 + 4 {}_F5(\xi) \eta^2 \\
& - 2 {}_F5(\xi)), \frac{d^3}{d\xi^3} {}_F5(\xi) = \left(- {}_F5(\xi)^2 R^2 W' \xi^5 + \left(\frac{d^2}{d\xi^2} {}_F5(\xi) \right) \left(\frac{d}{d\xi} \right. \right. \\
& \left. {}_F5(\xi) \right) \xi^6 + 2 {}_F5(\xi)^2 R^2 W' \xi^3 - 4 \left(\frac{d^2}{d\xi^2} {}_F5(\xi) \right) {}_F5(\xi) \xi^5 + 2 \left(\frac{d}{d\xi} {}_F5(\xi) \right)^2 \xi^5 \\
& - 2 {}_F5(\xi)^2 R \xi^4 - 3 \left(\frac{d^2}{d\xi^2} {}_F5(\xi) \right) \left(\frac{d}{d\xi} {}_F5(\xi) \right) \xi^4 - 2 \left(\frac{d}{d\xi} {}_F5(\xi) \right) {}_F5(\xi) \xi^4 \\
& - {}_F5(\xi)^2 R^2 W' \xi + 8 \left(\frac{d^2}{d\xi^2} {}_F5(\xi) \right) {}_F5(\xi) \xi^3 - 4 \left(\frac{d}{d\xi} {}_F5(\xi) \right)^2 \xi^3 \\
& + 4 {}_F5(\xi)^2 R \xi^2 + 3 \left(\frac{d^2}{d\xi^2} {}_F5(\xi) \right) \left(\frac{d}{d\xi} {}_F5(\xi) \right) \xi^2 + 4 \left(\frac{d}{d\xi} {}_F5(\xi) \right) {}_F5(\xi) \xi^2
\end{aligned}$$

$$\begin{aligned}
& + 2 _F5(\xi)^2 _c_3 \xi - 4 \left(\frac{d^2}{d\xi^2} _F5(\xi) \right) _F5(\xi) \xi + 2 \left(\frac{d}{d\xi} _F5(\xi) \right)^2 \xi - 2 _F5(\xi)^2 R \\
& - \left(\frac{d^2}{d\xi^2} _F5(\xi) \right) \left(\frac{d}{d\xi} _F5(\xi) \right) - 2 \left(\frac{d}{d\xi} _F5(\xi) \right) _F5(\xi) \left(\frac{d}{d\xi} _F5(\xi) \right) \Big/ \left(_F5(\xi) \xi^6 \right. \\
& \left. - 3 _F5(\xi) \xi^4 + 3 _F5(\xi) \xi^2 - _F5(\xi) \right) \Bigg]
\end{aligned}$$

by separation of variables, to obtain three ordinary differential equations in a product, which we separate.

```

> if has(op(1, op(1, op(2, solSeq))), phi) then
    Phieq := op(1, op(1, op(2, solSeq)));
  elif has(op(2, op(1, op(2, solSeq))), phi) then
    Phieq := op(2, op(1, op(2, solSeq)));
  else Phieq := op(3, op(1, op(2, solSeq)));
end if:
Phieq := subs(_F3 = Phi, Phieq);

```

$$Phieq := \frac{d^2}{d\phi^2} \Phi(\phi) = _c_3 \Phi(\phi) \quad (2.38)$$

```

> if has(op(1, op(1, op(2, solSeq))), eta) then
    Etaeq := op(1, op(1, op(2, solSeq)));
  elif has(op(2, op(1, op(2, solSeq))), eta) then
    Etaeq := op(2, op(1, op(2, solSeq)));
  else Etaeq := op(3, op(1, op(2, solSeq)));
end if:
Etaeq := subs(_F6 = Eta, _F5=Xi, Etaeq);

```

$$Etaeq := \frac{d^2}{d\eta^2} H(\eta) = \left(-2 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) + \Xi(\xi) H(\eta) R^2 W' \eta^4 \right. \quad (2.39)$$

$$\begin{aligned}
& - \Xi(\xi) H(\eta) R^2 W' \xi^4 + 4 \Xi(\xi) H(\eta) R \eta^2 \xi^3 - \Xi(\xi) H(\eta) R^2 W' \eta^2 \\
& + \Xi(\xi) H(\eta) R^2 W' \xi^2 - 4 \Xi(\xi) H(\eta) R \eta^2 \xi - 4 \Xi(\xi) \left(\frac{d}{d\eta} H(\eta) \right) \eta + 2 \left(\frac{d^2}{d\xi^2} \right. \\
& \Xi(\xi) \left. \right) H(\eta) \eta^2 - 2 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) \xi^4 + 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) \xi^2 + 4 \Xi(\xi) \left(\frac{d}{d\eta} \right. \\
& H(\eta) \left. \right) \eta^3 - 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) \xi^3 + 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) \xi - \Xi(\xi) H(\eta) R^2 W' \eta^4 \xi^2
\end{aligned}$$

$$\begin{aligned}
& + \Xi(\xi) H(\eta) R^2 W' \eta^2 \xi^4 + 4 \Xi(\xi) \left(\frac{d}{d\eta} H(\eta) \right) \eta \xi^2 - 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) \eta^2 \xi \\
& + 4 \Xi(\xi) H(\eta) R \xi - 4 \Xi(\xi) \left(\frac{d}{d\eta} H(\eta) \right) \eta^3 \xi^2 + 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) H(\eta) \eta^2 \xi^3 \\
& - 4 \Xi(\xi) H(\eta) R \xi^3 + 2 \Xi(\xi) H(\eta) \eta^2_{-c_3} - 2 \Xi(\xi) H(\eta) \xi^2_{-c_3} + 2 \left(\frac{d^2}{d\xi^2} \right. \\
& \left. \Xi(\xi) \right) H(\eta) \eta^2 \xi^4 - 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) H(\eta) \eta^2 \xi^2 \Bigg) / (2 \Xi(\xi) \eta^4 \xi^2 - 4 \Xi(\xi) \eta^2 \xi^2 \\
& - 2 \Xi(\xi) \eta^4 + 2 \Xi(\xi) \xi^2 + 4 \Xi(\xi) \eta^2 - 2 \Xi(\xi))
\end{aligned}$$

```

> if has(lhs(op(1, op(1, op(2, solSeq)))), xi) then
    Xieq := op(1, op(1, op(2, solSeq)));
    elif has(lhs(op(2, op(1, op(2, solSeq)))), xi) then
    Xieq := op(2, op(1, op(2, solSeq)));
    else Xieq := op(3, op(1, op(2, solSeq)));
end if:
Xieq := subs(_F5 = Xi, _F6 = Eta, Xieq);

```

$$\begin{aligned}
Xieq := \frac{d^3}{d\xi^3} \Xi(\xi) = \frac{1}{\Xi(\xi) \xi^6 - 3 \Xi(\xi) \xi^4 + 3 \Xi(\xi) \xi^2 - \Xi(\xi)} & \left(-\Xi(\xi)^2 R^2 W' \xi^5 + \left(\frac{d^2}{d\xi^2} \right. \right. \\
\Xi(\xi) \Bigg) \left(\frac{d}{d\xi} \Xi(\xi) \right) \xi^6 + 2 \Xi(\xi)^2 R^2 W' \xi^3 - 4 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) \Xi(\xi) \xi^5 + 2 \left(\frac{d}{d\xi} \Xi(\xi) \right)^2 \xi^5 & \\
- 2 \Xi(\xi)^2 R \xi^4 - 3 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) \left(\frac{d}{d\xi} \Xi(\xi) \right) \xi^4 - 2 \left(\frac{d}{d\xi} \Xi(\xi) \right) \Xi(\xi) \xi^4 & \\
- \Xi(\xi)^2 R^2 W' \xi + 8 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) \Xi(\xi) \xi^3 - 4 \left(\frac{d}{d\xi} \Xi(\xi) \right)^2 \xi^3 + 4 \Xi(\xi)^2 R \xi^2 & \\
+ 3 \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) \left(\frac{d}{d\xi} \Xi(\xi) \right) \xi^2 + 4 \left(\frac{d}{d\xi} \Xi(\xi) \right) \Xi(\xi) \xi^2 + 2 \Xi(\xi)^2_{-c_3} \xi - 4 \left(\frac{d^2}{d\xi^2} \right. & \\
\Xi(\xi) \Bigg) \Xi(\xi) \xi + 2 \left(\frac{d}{d\xi} \Xi(\xi) \right)^2 \xi - 2 \Xi(\xi)^2 R - \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) \left(\frac{d}{d\xi} \Xi(\xi) \right) - 2 \left(\frac{d}{d\xi} \right. & \\
\Xi(\xi) \Bigg) \Xi(\xi) \Bigg) &
\end{aligned} \tag{2.40}$$

Because the variables in the ordinary-differential equations for $\Xi(\xi)$ and $H(\eta)$ are incompletely separated we try to solve those two equations in a system.

```
> sol2 := dsolve({Xieq, Etaeq}, {Xi(xi), Eta(eta)});
sol2 := ( )
```

(2.41)

```
> Xisol := op(subs(sqrt(-_c[3])=m, sqrt(_c[3])=m, _c[3]=m^2, op(1,
sol2)));
Xisol := 1
```

(2.42)

```
> Etasol := op(subs(sqrt(-_c[3])=m, sqrt(_c[3])=m, _c[3]=m^2, op(2,
sol2)));
Etasol := 2
```

(2.43)

```
> #Etasol2 := dsolve(Etasol, Eta(eta));      # no answer produced
No solution is found for H(η).
```

```
> Xisol := dsolve(Xieq, Xi(xi));
```

$$Xisol := \Xi(\xi) = {}_2C_2 e^{\frac{R\sqrt{-2W'}\xi}{2}} \text{HeunC}\left(2R\sqrt{-2W'}, \sqrt{-c_3}, \sqrt{-c_3}, 4R, \frac{1}{2}R^2W' - 2R\right. \\ \left.+ {}_2C_1 - \frac{1}{2}c_3, \frac{\xi}{2} + \frac{1}{2}\right) (\xi^2 - 1)^{\frac{1}{2}\sqrt{-c_3}} + {}_2C_3 e^{\frac{R\sqrt{-2W'}\xi}{2}} \text{HeunC}\left(2R\sqrt{-2W'}, \right. \\ \left.-\sqrt{-c_3}, \sqrt{-c_3}, 4R, \frac{1}{2}R^2W' - 2R + {}_2C_1 - \frac{1}{2}c_3, \frac{\xi}{2} + \frac{1}{2}\right) \left(\frac{\xi}{2} + \frac{1}{2}\right) \\ \left.-\frac{1}{2}\sqrt{-c_3}\right) \left(\frac{\xi}{2} - \frac{1}{2}\right)^{\frac{1}{2}\sqrt{-c_3}}$$
(2.44)

Again the ordinary-differential equation for $\Xi(\xi)$ has an explicit algebraic solution, in terms of HeunC functions, but the ordinary-differential equation for $H(\eta)$ is not solved and still contains derivatives of HeunC functions of ξ . We solve the ordinary-differential equation for $\Phi(\phi)$.

```
> Phisol := dsolve(Phieq, Phi(phi));
```

$$Phisol := \Phi(\phi) = {}_2C_1 e^{\sqrt{-c_3}\phi} + {}_2C_2 e^{-\sqrt{-c_3}\phi}$$
(2.45)

```
> Phisol := subs(sqrt(_c[3])=m, _C2=0, _C1=1/sqrt(2*Pi), Phisol);
```

$$Phisol := \Phi(\phi) = \frac{\sqrt{2} e^{m\phi}}{2\sqrt{\pi}}$$
(2.46)

This result is the same as for the H atom in various systems of coordinates that include ϕ . The rest of the solution is intractable.

```
> restart:
```

In terms of these confocal elliptical coordinates, and following Pauling and Wilson, and Scott et alii, we separate the above partial-differential equation into three ordinary differential equations, in atomic units such that $m_e = 1$,

$$\psi(\eta, \xi, \phi) = H(\eta) \Xi(\xi) \Phi(\phi)$$

as follows. Equation $\Phi(\phi)$

$$\frac{d^2}{d\phi^2} \Phi(\phi) = -c^3 \Phi(\phi)$$

has, as above, normalized solution

$$\Phi(\phi) = \frac{e^{m\phi I}}{\sqrt{2\pi}}$$

in which m is the magnetic quantum number that appears also in the other two ordinary-differential equations. Here A is a separation parameter denoted μ by Pauling and Wilson, hence in their usage no longer denoting a reduced mass, and p , proportional to $\sqrt{W'}$, is denoted $\sqrt{\lambda}$ by Pauling and Wilson.

$$\begin{aligned} \frac{d}{d\eta} \left((1 - \eta^2) \left(\frac{d}{d\eta} H(\eta) \right) \right) + \left(-\frac{m^2}{1 - \eta^2} + p^2 \eta^2 - q_1 \eta - A \right) H(\eta) &= 0 \\ \frac{d}{d\xi} \left((\xi^2 - 1) \left(\frac{d}{d\xi} \Xi(\xi) \right) \right) + \left(-\frac{m^2}{\xi^2 - 1} - p^2 \xi^2 + q_2 \xi + A \right) \Xi(\xi) &= 0 \end{aligned}$$

In these ordinary differential equations appear separation parameter $A = \mu$ in the former, and μ ,

$$\begin{aligned} p^2 = -\frac{W'R^2}{2} = \lambda \text{ in the latter is proportional to electronic energy } W' = W - \frac{1}{R}, q_1 = R(Z_A - Z_B), q_2 \\ = R(Z_A + Z_B); \text{ for } {}^1\text{H}_2^+, q_1 = 0 \text{ and } q_2 = 2R. \text{ In the limit as } R \rightarrow 0, \lim_{R \rightarrow 0} (A = -l(l+1)) \text{ and} \\ \lim_{R \rightarrow 0} (p^2 = 0). \end{aligned}$$

$$\begin{aligned} &> \text{Etaeq} := \text{diff}((1-\text{eta}^2)*\text{diff}(\text{Eta}(\text{eta}),\text{eta}),\text{eta}) + (-m^2/(1-\text{eta}^2)+ \\ &\quad p^2*\text{eta}^2-q[1]*\text{eta}-A)*\text{Eta}(\text{eta}) = 0; \end{aligned}$$

$$\begin{aligned} \text{Etaeq} := -2\eta \left(\frac{d}{d\eta} H(\eta) \right) + (-\eta^2 + 1) \left(\frac{d^2}{d\eta^2} H(\eta) \right) + \left(-\frac{m^2}{-\eta^2 + 1} + p^2 \eta^2 - q_1 \eta \right. \\ \left. - A \right) H(\eta) = 0 \end{aligned} \quad (2.47)$$

$$\begin{aligned} &> \text{Xieq} := \text{diff}((\text{xi}^2-1)*\text{diff}(\text{Xi}(\text{xi}),\text{xi}),\text{xi}) + (-m^2/(\text{xi}^2-1)-p^2* \\ &\quad \text{xi}^2+q[2]*\text{xi}+A)*\text{Xi}(\text{xi}) = 0; \end{aligned}$$

$$\begin{aligned} \text{Xieq} := 2\xi \left(\frac{d}{d\xi} \Xi(\xi) \right) + (\xi^2 - 1) \left(\frac{d^2}{d\xi^2} \Xi(\xi) \right) + \left(-\frac{m^2}{\xi^2 - 1} - p^2 \xi^2 + q_2 \xi + A \right) \Xi(\xi) \\ = 0 \end{aligned} \quad (2.48)$$

$$> \text{diff}(\text{Phi}(\text{phi}),\text{phi}^2) = -c^3*\text{Phi}(\text{phi});$$

$$\frac{d^2}{d\phi^2} \Phi(\phi) = -c^3 \Phi(\phi) \quad (2.49)$$

$$> \text{Etasol} := \text{dsolve}(\text{Etaeq}, \text{Eta}(\text{eta}));$$

$$\text{Etasol} := H(\eta) = {}_1F_1 \left(4p, m, m, 2q_1, \frac{m^2}{2} - p^2 + A - q_1, \frac{\eta}{2} + \frac{1}{2} \right) (\eta^2 - 1)^{\frac{m}{2}} \quad (2.50)$$

$$+ _C2 e^{p\eta} \text{HeunC}\left(4p, -m, m, 2q_1, \frac{m^2}{2} - p^2 + A - q_1, \frac{\eta}{2} + \frac{1}{2}\right) \sqrt{2\eta - 2} (\eta + 1)^{-\frac{m}{2}} (\eta - 1)^{\frac{m}{2} - \frac{1}{2}}$$

> Xisol := dsolve(Xieq, Xi(xi));

$$Xisol := \Xi(\xi) = _C1 e^{\xi p} \text{HeunC}\left(4p, m, m, 2q_2, \frac{m^2}{2} - p^2 + A - q_2, \frac{\xi}{2} + \frac{1}{2}\right) (\xi^2 - 1)^{\frac{m}{2}} \quad (2.51)$$

$$+ _C2 e^{\xi p} \text{HeunC}\left(4p, -m, m, 2q_2, \frac{m^2}{2} - p^2 + A - q_2, \frac{\xi}{2} + \frac{1}{2}\right) \sqrt{2\xi - 2} (\xi + 1)^{-\frac{m}{2}} (\xi - 1)^{\frac{m}{2} - \frac{1}{2}}$$

We eliminate unphysical terms.

> Etasol := subs(_C2=0, Etasol);

$$Etasol := H(\eta) = _C1 e^{p\eta} \text{HeunC}\left(4p, m, m, 2q_1, \frac{m^2}{2} - p^2 + A - q_1, \frac{\eta}{2} + \frac{1}{2}\right) (\eta^2 - 1)^{\frac{m}{2}} \quad (2.52)$$

> Xisol := subs(_C2=0, Xisol);

$$Xisol := \Xi(\xi) = _C1 e^{\xi p} \text{HeunC}\left(4p, m, m, 2q_2, \frac{m^2}{2} - p^2 + A - q_2, \frac{\xi}{2} + \frac{1}{2}\right) (\xi^2 - 1)^{\frac{m}{2}} \quad (2.53)$$

> Phisol := Phi(phi) = exp(I*m*phi)/sqrt(2*Pi);

$$Phisol := \Phi(\phi) = \frac{e^{Im\phi} \sqrt{2}}{2\sqrt{\pi}} \quad (2.54)$$

The total solution therefore becomes

> psi := rhs(Xisol)*rhs(Etasol)*rhs(Phisol);

$$\psi := \frac{1}{2\sqrt{\pi}} \left(_C1^2 e^{\xi p} \text{HeunC}\left(4p, m, m, 2q_2, \frac{m^2}{2} - p^2 + A - q_2, \frac{\xi}{2} + \frac{1}{2}\right) (\xi^2 - 1)^{\frac{m}{2}} e^{p\eta} \text{HeunC}\left(4p, m, m, 2q_1, \frac{m^2}{2} - p^2 + A - q_1, \frac{\eta}{2} + \frac{1}{2}\right) (\eta^2 - 1)^{\frac{m}{2}} e^{Im\phi} \sqrt{2} \right) \quad (2.55)$$

with one normalizing factor, N .

> psi := subs(_C1^2=N, psi);

$$\psi := \frac{1}{2\sqrt{\pi}} \left(N e^{\xi p} \text{HeunC}\left(4p, m, m, 2q_2, \frac{m^2}{2} - p^2 + A - q_2, \frac{\xi}{2} + \frac{1}{2}\right) (\xi^2 - 1)^{\frac{m}{2}} e^{p\eta} \text{HeunC}\left(4p, m, m, 2q_1, \frac{m^2}{2} - p^2 + A - q_1, \frac{\eta}{2} + \frac{1}{2}\right) (\eta^2 - 1)^{\frac{m}{2}} e^{Im\phi} \sqrt{2} \right) \quad (2.56)$$

$$-1) \frac{m}{2} e^{p\eta} \text{HeunC}\left(4p, m, m, 2q_1, \frac{m^2}{2} - p^2 + A - q_1, \frac{\eta}{2} + \frac{1}{2}\right) (\eta^2 - 1)^{\frac{m}{2}} e^{Im\phi\sqrt{2}} \right)$$

Hence $\psi(x, y, z) \rightarrow \psi(\xi, \eta, \phi) = \Xi(\xi) H(\eta) \Phi(\phi)$, and the amplitude equation has been assumed to become separable into three equations, of which the solutions appear above. Parameters m , λ and separation parameter A must assume characteristic values for the equations to possess acceptable solutions. The subsequent procedure involves finding a relation between λ and A such that the equation for η possesses a satisfactory solution, and then, with the use of that relation, finding from the ξ equation the characteristic values of λ and hence energies W' and W . The **HeunC** functions remain intractable.

3.63 matrices and their determinants for $H_2^{'+}$

> restart:

For $^1H_2^{'+}$, we recall three ordinary differential equations for the solution of the total amplitude function as a product of functions of separate variables in the coordinate system confocal elliptical or prolate spheroidal, $\psi(\eta, \xi, \phi) = H(\eta) \Xi(\xi) \Phi(\phi)$, as follows,

$$\begin{aligned} \frac{d}{d\eta} \left((1 - \eta^2) \left(\frac{d}{d\eta} H(\eta) \right) \right) + \left(-\frac{m^2}{1 - \eta^2} + p^2 \eta^2 - q_1 \eta - A \right) H(\eta) &= 0 \\ \frac{d}{d\xi} \left((\xi^2 - 1) \left(\frac{d}{d\xi} \Xi(\xi) \right) \right) + \left(-\frac{m^2}{\xi^2 - 1} - p^2 \xi^2 + q_2 \xi + A \right) \Xi(\xi) &= 0 \\ \frac{d^2}{d\phi^2} \Phi(\phi) &= -c^2 \Phi(\phi) \end{aligned}$$

in which appear separation parameter A related to the total angular momentum, $p^2 = -\frac{W'R^2}{2} = \lambda$ is

proportional to electronic energy W' , $q_1 = R(Z_A - Z_B)$, $q_2 = R(Z_A + Z_B)$; for $^1H_2^{'+}$, $q_1 = 0$ and $q_2 = 2R$. In the limit as $R \rightarrow 0$, $\lim_{R \rightarrow 0} (A = -l(l+1))$ and $\lim_{R \rightarrow 0} (p^2 = 0)$. Internuclear distance

R remains, explicitly or implicitly, a parameter of the system. For these equations to possess acceptable solutions, parameters m , λ and separation parameter A must assume characteristic values. Just as for the H atom, parameter m serves as a magnetic quantum number pertaining to a component of total angular momentum about the internuclear axis, assuming integer values, $m = 0, \pm 1, \pm 2, \dots$.

To solve this system, we express $M(\eta, \phi) = H(\eta) \Phi(\phi)$ in an expansion of spherical harmonics $Y_{k,m}(\eta, \phi)$,

$$M(\eta, \phi) = \sum_{k=m}^{\infty} f_{k,m} Y_{k,m}(\eta, \phi)$$

Injection of this basis into the ordinary differential equation for $M(\eta, \phi)$ or $H(\eta)$ and $\Phi(\phi)$ produces a symmetric matrix **F** of which the determinant must vanish when p and A satisfy the eigenvalue equation $H\psi = W\psi$. We form this matrix with this procedure.

```
> Fm := proc(N)
  local i,F;
```

```

global p,A,m,Q1,R;
F := Matrix(N,N, shape=symmetric);
for i to N do
  F[i,i] := -i*(i+1) +p^2*((2*i^2-2*m^2+2*i-1)/((2*i+3)*(2*i-1))
) - A;
end do;
for i to N-1 do
  F[i+1,i] := -R*Q1*((i+m+1)*(i-m+1)/((2*i+3)*(2*i+1)))^(1/2);
end do;
for i to N-2 do
  F[i+2,i] := p^2/(2*i+3)*((i+m+1)*(i-m+1)*(i+m+2)*(i-m+2)/((2*
i+5)*(2*i+1)))^(1/2);
end do;
F;
end proc:

```

For a general diatomic molecule with one electron, we form this matrix of order 6,

> Fmm := Fm(6);

$$\begin{aligned}
 Fmm := & \left[\left[-2 + \frac{p^2(-2m^2+3)}{5} - A, -\frac{RQ1\sqrt{15}\sqrt{(2+m)(2-m)}}{15}, \right. \right. \\
 & \left. \frac{p^2\sqrt{21}\sqrt{(2+m)(2-m)(3+m)(3-m)}}{105}, 0, 0, 0 \right], \\
 & \left[-\frac{RQ1\sqrt{15}\sqrt{(2+m)(2-m)}}{15}, -6 + \frac{p^2(-2m^2+11)}{21} - A, \right. \\
 & \left. -\frac{RQ1\sqrt{35}\sqrt{(3+m)(3-m)}}{35}, \frac{p^2\sqrt{45}\sqrt{(3+m)(3-m)(4+m)(4-m)}}{315}, 0, 0 \right], \\
 & \left[\frac{p^2\sqrt{21}\sqrt{(2+m)(2-m)(3+m)(3-m)}}{105}, -\frac{RQ1\sqrt{35}\sqrt{(3+m)(3-m)}}{35}, -12 \right. \\
 & \left. + \frac{p^2(-2m^2+23)}{45} - A, -\frac{RQ1\sqrt{63}\sqrt{(4+m)(4-m)}}{63}, \right. \\
 & \left. \frac{p^2\sqrt{77}\sqrt{(4+m)(4-m)(5+m)(5-m)}}{693}, 0 \right], \\
 & \left[0, \frac{p^2\sqrt{45}\sqrt{(3+m)(3-m)(4+m)(4-m)}}{315}, -\frac{RQ1\sqrt{63}\sqrt{(4+m)(4-m)}}{63}, \right. \\
 & \left. -20 + \frac{p^2(-2m^2+39)}{77} - A, -\frac{RQ1\sqrt{99}\sqrt{(5+m)(5-m)}}{99}, \right. \\
 & \left. \frac{p^2\sqrt{117}\sqrt{(5+m)(5-m)(6+m)(6-m)}}{1287} \right],
 \end{aligned} \tag{3.1}$$

$$\left[0, 0, \frac{p^2 \sqrt{77} \sqrt{(4+m)(4-m)(5+m)(5-m)}}{693}, -\frac{R Q I \sqrt{99} \sqrt{(5+m)(5-m)}}{99}, \right. \\ \left. -30 + \frac{p^2 (-2m^2 + 59)}{117} - A, -\frac{R Q I \sqrt{143} \sqrt{(6+m)(6-m)}}{143} \right], \\ \left[0, 0, 0, \frac{p^2 \sqrt{117} \sqrt{(5+m)(5-m)(6+m)(6-m)}}{1287}, \right. \\ \left. -\frac{R Q I \sqrt{143} \sqrt{(6+m)(6-m)}}{143}, -42 + \frac{p^2 (-2m^2 + 83)}{165} - A \right] \Bigg]$$

which shows a pentadiagonal form. We evaluate its determinant.

> DFm := LinearAlgebra:-Determinant(Fmm);

$$\begin{aligned} DFm := & 3628800 + 3110400 A + \frac{1531312 m^6 p^2 Q I^4 R^4}{156080925} - \frac{185734815092 m^2 p^6 R^2 Q I^2}{18261468225} \\ & - \frac{21444832 m^4 p^2 R^4 Q I^4}{52026975} + \frac{212184604 m^2 p^2 R^4 Q I^4}{52026975} + \frac{2321344664 m^4 p^6 R^2 Q I^2}{1217431215} \\ & + \frac{1528928 m^8 p^6 Q I^2 R^2}{730458729} - \frac{2117785136 m^6 p^6 R^2 Q I^2}{18261468225} \\ & - \frac{1}{18393375} (8 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\ & \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^4 R^4 p^4) \\ & + \frac{1}{22347950625} (9794 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\ & \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^2 R^2 p^8) \\ & + \frac{1}{2269176525} (166 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\ & \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 p^8) \\ & - \frac{1}{86089778775} (2746 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\ & \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^4 R^4 p^4) \\ & + \frac{1}{730458729} (10 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\ & \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 p^8) \\ & + \frac{1}{429975} (21272 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^4 \\ & \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)}) \end{aligned} \tag{3.2}$$

$$\begin{aligned}
& - \frac{1}{135442125} (105832 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} \\
& p^6 \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)}) \\
& + \frac{1}{978193125} (3422 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^8 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)}) \\
& - \frac{1}{1260653625} (11642 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} \\
& p^4 \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)}) \\
& + \frac{1}{429975} (244 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^2 \sqrt{21} \\
& \sqrt{(2+m)(-2+m)(3+m)(-3+m)}) \\
& - \frac{1}{35} (32 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^2 \sqrt{21} \\
& \sqrt{(2+m)(-2+m)(3+m)(-3+m)}) \\
& + \frac{1}{3087315} (16 R^4 Q I^4 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} p^2 \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)}) \\
& - \frac{1}{10697546475} (664 R^4 Q I^4 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} p^4 \\
& \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)}) \\
& - \frac{1}{382621239} (1328 m^2 p^4 R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& - \frac{1}{674817} (8 A R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} p^2 \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& - \frac{1}{33960465} (1712 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} p^4 \\
& \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} m^2) \\
& - \frac{1}{1440747} (232 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} p^2 \\
& \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} A)
\end{aligned}$$

$$\begin{aligned}
& - \frac{1}{3443591151} \left(212 m^4 p^6 R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \right. \\
& \left. \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} \right) \\
& - \frac{1}{74882825325} \left(56528 m^4 p^6 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \right. \\
& \left. \sqrt{-(5+m)(-5+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \right) \\
& + \frac{1}{3443591151} \left(1732 m^2 p^6 R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \right. \\
& \left. \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} \right) \\
& + \frac{1}{74882825325} \left(435184 m^2 p^6 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \right. \\
& \left. \sqrt{-(5+m)(-5+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \right) \\
& + \frac{1}{163980531} \left(400 p^4 A R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \right. \\
& \left. \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} \right) \\
& + \frac{1}{4992188355} \left(129064 p^4 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \right. \\
& \left. \sqrt{-(5+m)(-5+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} A \right) \\
& - \frac{1}{18220059} \left(40 A^2 R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} p^2 \sqrt{117} \right. \\
& \left. \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} \right) \\
& - \frac{1}{4322241} \left(100 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} p^2 \right. \\
& \left. \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} A^2 \right) \\
& - \frac{1}{243795825} \left(304 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right. \\
& \left. \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} m^4 p^6 \right) \\
& - \frac{1}{99324225} \left(4 R^4 Q I^4 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right. \\
& \left. \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} m^2 p^2 \right) \\
& + \frac{1}{1031443875} \left(45104 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right. \\
& \left. \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} m^2 p^6 \right)
\end{aligned}$$

$$\begin{aligned}
& - \frac{1}{297972675} (434288 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} m^2 p^4) \\
& + \frac{1}{638512875} (149816 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} A \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} p^4) \\
& - \frac{1}{694575} (148 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} A^2 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} p^2) \\
& - \frac{1}{25725} (104 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} A \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} p^2) \\
& - \frac{1}{42513471} (4 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^4 R^4 m^2 p^2) \\
& - \frac{1}{67043851875} (16 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^2 R^2 m^6 p^8) \\
& - \frac{1}{74882825325} (16 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 m^6 p^8) \\
& - \frac{1}{24105138057} (4 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 m^6 p^8) \\
& - \frac{1}{496621125} (4 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^4 R^4 m^4 p^4) \\
& + \frac{1}{13408770375} (232 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^2 R^2 m^4 p^8) \\
& + \frac{1}{74882825325} (776 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 m^4 p^8)
\end{aligned}$$

$$\begin{aligned}
& - \frac{1}{86089778775} (136 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} QI^4 R^4 m^4 p^4) \\
& + \frac{1}{24105138057} (74 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} QI^2 R^2 m^4 p^8) \\
& + \frac{1}{6621615} (2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} QI^4 R^4 m^2 p^4) \\
& - \frac{1}{67043851875} (21292 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} QI^2 R^2 m^2 p^8) \\
& - \frac{1}{14976565065} (956 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} QI^2 R^2 m^2 p^8) \\
& + \frac{1}{12298539825} (224 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} QI^4 R^4 m^2 p^4) \\
& - \frac{1}{24105138057} (346 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} QI^2 R^2 m^2 p^8) \\
& + \frac{1}{11036025} (8 A \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} QI^4 R^4 p^2) \\
& - \frac{1}{5157219375} (12742 A \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} QI^2 R^2 p^6) \\
& - \frac{1}{1528220925} (622 A \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} QI^2 R^2 p^6) \\
& + \frac{1}{34783749} (2 A \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} QI^4 R^4 p^2)
\end{aligned}$$

$$\begin{aligned}
& - \frac{1}{3443591151} (310 A \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 p^6) \\
& + \frac{1}{4469590125} (20686 A^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^2 R^2 p^4) \\
& + \frac{1}{4992188355} (3758 A^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 p^4) \\
& + \frac{1}{1147863717} (206 A^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 p^4) \\
& - \frac{1}{694575} (2 A^3 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^2 R^2 p^2) \\
& - \frac{1}{4322241} (2 A^3 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 p^2) \\
& - \frac{1}{18220059} (2 A^3 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 p^2) \\
& - \frac{1}{26411214375} (4 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^8 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^6) \\
& + \frac{1}{26411214375} (362 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^8 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^4) \\
& - \frac{1}{5282242875} (2066 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^8 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^2) \\
& - \frac{1}{96744375} (2458 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^6 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} A)
\end{aligned}$$

$$\begin{aligned}
& + \frac{1}{225736875} (12398 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^4 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} A^2) \\
& - \frac{1}{81942485625} (904 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^4 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^4) \\
& + \frac{1}{81942485625} (54176 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} \\
& p^4 \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^2) \\
& - \frac{1}{245} (24 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^2 \sqrt{21} \\
& \sqrt{(2+m)(-2+m)(3+m)(-3+m)} A) \\
& - \frac{1}{55125} (2 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^2 \sqrt{21} \\
& \sqrt{(2+m)(-2+m)(3+m)(-3+m)} A^3) \\
& - \frac{1}{2149875} (44 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^2 \sqrt{21} \\
& \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^2) \\
& + \frac{1}{6449625} (118 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^2 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} A) \\
& - \frac{1}{429975} (848 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^4 \sqrt{21} \\
& \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^2) \\
& - \frac{1}{135442125} (148 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^6 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^4) \\
& + \frac{1}{135442125} (8132 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^6 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^2) \\
& + \frac{1}{45147375} (152144 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^4 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} A)
\end{aligned}$$

$$\begin{aligned}
& - \frac{1}{55125} (184 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^2 \sqrt{21} \\
& \sqrt{(2+m)(-2+m)(3+m)(-3+m)} A^2) - \frac{22832640 p^2}{13} \\
& - \frac{1}{2678348673} (8 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} p^4 \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& - \frac{1}{163980531} (160 m^2 p^4 A R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& - \frac{1}{4992188355} (54256 m^2 p^4 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \\
& \sqrt{-(5+m)(-5+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} A) \\
& + \frac{1}{843679831995} (976 p^6 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& + \frac{1}{124804708875} (8648 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} p^6) \\
& - \frac{1}{36018675} (8 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} p^4) \\
& - \frac{1}{4469590125} (385328 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} A \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} m^2 p^4) \\
& - \frac{1}{9577693125} (328 A \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^2 R^2 m^4 p^6)
\end{aligned}$$

$$\begin{aligned}
& - \frac{1}{74882825325} (1528 A \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 m^4 p^6) \\
& - \frac{1}{313053741} (2 A \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 m^4 p^6) \\
& - \frac{1}{208007848125} (166 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 p^8) \\
& - \frac{1}{25566055515} (2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \\
& \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 p^8) \\
& - \frac{1}{99324225} (2 A \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^4 R^4 m^2 p^2) \\
& + \frac{1}{67043851875} (84008 A \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^2 R^2 m^2 p^6) \\
& + \frac{1}{74882825325} (18344 A \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 m^2 p^6) \\
& - \frac{1}{382621239} (4 A \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^4 R^4 m^2 p^2) \\
& + \frac{1}{3443591151} (206 A \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 m^2 p^6) \\
& - \frac{1}{4469590125} (5524 A^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} Q I^2 R^2 m^2 p^4)
\end{aligned}$$

$$\begin{aligned}
& - \frac{1}{4992188355} (1172 A^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 m^2 p^4) \\
& - \frac{1}{1147863717} (68 A^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 m^2 p^4) \\
& - \frac{1}{45147375} (5792 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^4 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^2 A) \\
& - \frac{1}{677210625} (22 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^6 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^4 A) \\
& + \frac{1}{677210625} (1262 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^6 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^2 A) \\
& - \frac{1}{225736875} (452 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^4 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} m^2 A^2) \\
& - \frac{1}{6449625} (4 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^2 \sqrt{21} \\
& \sqrt{(2+m)(-2+m)(3+m)(-3+m)} A m^2) \\
& - \frac{1}{99225} (8 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^4 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)}) \\
& + \frac{1}{2681754075} (6224 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^6 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)}) \\
& - \frac{1}{603394666875} (9794 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^8 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)}
\end{aligned}$$

$$\begin{aligned}
& \sqrt{63} \sqrt{-(4+m)(-4+m)} \\
& + \frac{1}{496621125} (8 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^4 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)}) \\
& - \frac{1}{64833615} (2 A \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^4 R^4 m^2 p^2) \\
& + \frac{1}{120525690285} (8 R^4 Q I^4 \sqrt{63} \sqrt{-(4+m)(-4+m)} p^4 \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& - \frac{1}{1406133053325} (4 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} p^6) \\
& + \frac{1}{2343555088875} (2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 p^8) \\
& - \frac{1}{124804708875} (136 A \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 m^2 p^6) \\
& - \frac{1}{64898448615} (8 A \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \\
& \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 m^2 p^6) \\
& - \frac{1}{201131555625} (376 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^6 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)}
\end{aligned}$$

$$\begin{aligned}
& \sqrt{63} \sqrt{-(4+m)(-4+m)} m^2 A) \\
& - \frac{1}{1620840375} (4 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^6 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)}) \\
& + \frac{1}{5616211899375} (166 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^8 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)}) \\
& - \frac{1}{3087315} (4 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^4 R^4 m^2 p^2) \\
& - \frac{1}{10697546475} (4 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^4 R^4 m^4 p^4) \\
& + \frac{1}{1528220925} (26 R^4 Q I^4 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} p^4 \\
& \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} m^2) \\
& + \frac{1}{64833615} (8 R^4 Q I^4 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} p^2 \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} A) \\
& - \frac{1}{5616211899375} (4 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^8 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} m^2) \\
& - \frac{1}{34037647875} (2 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^6 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} A) \\
& + \frac{5053 p^{12}}{920205} + A^6 - \frac{8 R^6 Q I^6 m^4}{19305} + \frac{112 R^6 Q I^6 m^2}{19305} + \frac{6701258 A^2 Q I^2 R^2 m^6 p^4}{2341213875}
\end{aligned}$$

$$\begin{aligned}
& - \frac{96284 A Q I^4 R^4 m^4 p^2}{5780775} + \frac{49266304 A Q I^2 R^2 m^4 p^6}{621138375} - \frac{6659504 A^2 Q I^2 R^2 m^4 p^4}{60031125} \\
& + \frac{5853304 A^3 Q I^2 R^2 m^4 p^2}{156080925} + \frac{28699156 A Q I^4 R^4 m^2 p^2}{156080925} - \frac{8512856728 A Q I^2 R^2 m^2 p^6}{18261468225} \\
& + \frac{73850908 A^2 Q I^2 R^2 m^2 p^4}{70945875} - \frac{128664572 A^3 Q I^2 R^2 m^2 p^2}{156080925} + \frac{616744 A Q I^2 R^2 m^8 p^6}{8300667375} \\
& + \frac{168968 A Q I^4 R^4 m^6 p^2}{468242775} - \frac{57847388 A Q I^2 R^2 m^6 p^6}{13043905875} - \frac{256 R^6 Q I^6}{15015} \\
& - \frac{3575102 A Q I^4 R^4 p^2}{8513505} + \frac{58701762272 A Q I^2 R^2 p^6}{91307341125} + \frac{5 A^4 Q I^2 R^2 m^2}{39} \\
& - \frac{366417644 A^2 Q I^2 R^2 p^4}{180093375} + \frac{422002624 A^3 Q I^2 R^2 p^2}{156080925} + \frac{54062 Q I^2 R^2 m^{10} p^8}{91307341125} \\
& + \frac{61976 Q I^4 R^4 m^8 p^4}{7023641625} - \frac{4561412 Q I^2 R^2 m^8 p^8}{91307341125} - \frac{4088792 Q I^4 R^4 m^6 p^4}{7023641625} \\
& + \frac{8658172 Q I^2 R^2 m^6 p^8}{6087156075} + \frac{26640046 Q I^4 R^4 m^4 p^4}{2341213875} - \frac{1484976488 Q I^2 R^2 m^4 p^8}{91307341125} \\
& + \frac{2 A^2 Q I^4 R^4 m^4}{715} - \frac{511026344 Q I^4 R^4 m^2 p^4}{7023641625} + \frac{6274544201 Q I^2 R^2 m^2 p^8}{91307341125} \\
& - \frac{12 A^2 Q I^4 R^4 m^2}{143} + \frac{16 m^{12} p^{12}}{4347968625} - \frac{1744 m^{10} p^{12}}{4347968625} + \frac{7652 m^8 p^{12}}{483107625} - \frac{1217512 m^6 p^{12}}{4347968625} \\
& + \frac{9660316 m^4 p^{12}}{4347968625} - \frac{9968248 m^2 p^{12}}{1449322875} - \frac{283886 A p^{10}}{2760615} + \frac{298877 A^2 p^8}{418275} \\
& - \frac{2276324 A^3 p^6}{975975} + \frac{24877 A^4 p^4}{6435} - \frac{614 A^5 p^2}{195} + \frac{1949155616 m^2 p^{10}}{869593725} \\
& - \frac{107913616 m^4 p^{10}}{173918745} + \frac{28352 m^{10} p^{10}}{869593725} - \frac{2141792 m^8 p^{10}}{869593725} + \frac{2574592 m^6 p^{10}}{41409225} \\
& - \frac{11536 p^2 A^4}{39} + \frac{1879792 p^4 A^3}{6435} + \frac{79519168 p^8 A}{2927925} - \frac{388497808 p^6 A^2}{2927925} + 112 A^5 \\
& - \frac{27054656 p^{10}}{13803075} + \frac{17569304 p^8}{75075} + 4648 A^4 + \frac{32807920 p^4}{143} - \frac{6618787712 p^6}{585585} \\
& + 89280 A^3 + \frac{116 A^5 m^2 p^2}{195} - \frac{50 A^4 Q I^2 R^2}{39} + \frac{Q I^6 R^6 m^6}{135135} + \frac{976 A m^{10} p^{10}}{869593725} \\
& - \frac{1208 A m^8 p^{10}}{13378365} + \frac{3044 A^2 m^8 p^8}{26351325} + \frac{101408 A m^6 p^{10}}{41409225} - \frac{296 A^2 m^6 p^8}{46475}
\end{aligned}$$

$$\begin{aligned}
& + \frac{43136 A^3 m^6 p^6}{8783775} - \frac{4612568 A m^4 p^{10}}{173918745} + \frac{26516 A^2 m^4 p^8}{250965} - \frac{207712 A^3 m^4 p^6}{1254825} \\
& + \frac{91582156 A m^2 p^{10}}{869593725} + \frac{568 A^4 m^4 p^4}{6435} - \frac{15065168 A^2 m^2 p^8}{26351325} + \frac{6250333 QI^4 R^4 p^4}{56189133} \\
& - \frac{201501254 QI^2 R^2 p^8}{2766889125} + \frac{1719752 A^3 m^2 p^6}{1254825} - \frac{1912 A^4 m^2 p^4}{1287} + \frac{283 A^2 QI^4 R^4}{715} \\
& + \frac{17629826924 p^6 R^2 QI^2}{1404728325} - \frac{257345636 p^2 R^4 QI^4}{31216185} - \frac{3820 A^3 R^2 QI^2}{39} + \frac{33148 A R^4 QI^4}{2145} \\
& + \frac{2464 m^2 p^2 A^4}{39} + \frac{53488 m^4 p^4 A^3}{6435} + \frac{167008 m^8 p^8 A}{26351325} - \frac{2855968 m^6 p^8 A}{8783775} \\
& + \frac{3312544 m^6 p^6 A^2}{8783775} + \frac{1244816 m^4 p^8 A}{250965} - \frac{14666768 m^4 p^6 A^2}{1254825} - \frac{162880 m^2 p^4 A^3}{1287} \\
& - \frac{641795296 m^2 p^8 A}{26351325} + \frac{109981408 m^2 p^6 A^2}{1254825} + \frac{20 QI^4 R^4 m^4}{11} + \frac{475328 m^2 p^2 A^3}{195} \\
& + \frac{8793536 m^6 p^6 A}{975975} - \frac{106293856 m^4 p^6 A}{418275} + \frac{1741936 m^4 p^4 A^2}{6435} + \frac{712994336 m^2 p^6 A}{418275} \\
& - \frac{23661776 m^2 p^4 A^2}{6435} - \frac{22197506888 p^4 R^2 QI^2}{31216185} - \frac{33704 A^2 R^2 QI^2}{13} + \frac{4104768 m^2 p^2 A}{13} \\
& + \frac{284976 QI^2 R^2 m^2}{13} + \frac{7656256 m^4 p^4 A}{2145} - \frac{91177088 m^2 p^4 A}{2145} + \frac{2729088 m^2 p^2 A^2}{65} \\
& + \frac{1485505136 p^2 R^2 QI^2}{99099} - \frac{361008 A R^2 QI^2}{13} - \frac{8454232924 m^2 p^2 A^2 R^2 QI^2}{156080925} \\
& - \frac{837559168 m^4 p^4 A R^2 QI^2}{156080925} + \frac{71269424 m^6 p^4 A QI^2 R^2}{468242775} + \frac{472170788 m^2 p^4 A R^2 QI^2}{10405395} \\
& + \frac{436798328 m^4 p^2 A^2 QI^2 R^2}{156080925} - \frac{6040 R^4 QI^4 m^2}{143} - \frac{11553555248 m^2 p^2 A R^2 QI^2}{10405395} \\
& + \frac{52555936 m^4 p^2 A QI^2 R^2}{800415} - \frac{2093791376 m^2 p^8}{8783775} + \frac{720704 m^8 p^8}{8783775} - \frac{11383808 m^6 p^8}{2927925} \\
& + \frac{22617968 m^4 p^8}{418275} - \frac{1931552 p^2 A^3}{195} - \frac{601398368 p^6 A}{266175} + \frac{47614768 p^4 A^2}{6435} \\
& + \frac{10460160 m^2 p^2}{13} + \frac{2265280 m^4 p^4}{143} - \frac{23134016 m^2 p^4}{143} - \frac{11541216 p^2 A}{13} + 808848 A^2 \\
& + \frac{38870144 m^6 p^6}{585585} - \frac{46917056 m^4 p^6}{27885} + \frac{277175168 m^2 p^6}{27885} + \frac{156343936 p^4 A}{2145}
\end{aligned}$$

$$\begin{aligned}
& - \frac{9387072 p^2 A^2}{65} \\
& - \frac{1}{120525690285} \left(2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \right. \\
& \quad \left. \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \right. \\
& \quad \left. \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^4 R^4 m^2 p^4 \right) \\
& - \frac{1}{624023544375} \left(8 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right. \\
& \quad \left. \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \right. \\
& \quad \left. \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 m^4 p^8 \right) \\
& - \frac{1}{843679831995} \left(8 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \right. \\
& \quad \left. \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \right. \\
& \quad \left. \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 m^4 p^8 \right) \\
& + \frac{1}{624023544375} \left(344 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right. \\
& \quad \left. \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \right. \\
& \quad \left. \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 m^2 p^8 \right) \\
& + \frac{1}{120525690285} \left(8 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \right. \\
& \quad \left. \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \right. \\
& \quad \left. \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 m^2 p^8 \right) \\
& + \frac{1}{17829244125} \left(52 A \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right. \\
& \quad \left. \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \right. \\
& \quad \left. \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 p^6 \right) \\
& + \frac{1}{843679831995} \left(236 A \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \right. \\
& \quad \left. \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \right. \\
& \quad \left. \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 p^6 \right) \\
& - \frac{1}{756392175} \left(2 A^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right.
\end{aligned}$$

$$\begin{aligned}
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} Q I^2 R^2 p^4) \\
& - \frac{1}{8035046019} (2 A^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \\
& \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 p^4) \\
& - \frac{1}{13408770375} (928 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^6 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)} m^2) \\
& - \frac{1}{3472875} (16 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^4 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)} A) \\
& - \frac{1}{603394666875} (8 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^8 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)} m^4) \\
& + \frac{1}{603394666875} (568 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^8 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)} m^2) \\
& + \frac{1}{28733079375} (1852 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^6 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)} A) \\
& - \frac{1}{31255875} (2 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^4 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)} A^2)
\end{aligned}$$

$$\begin{aligned}
& - \frac{1}{4469590125} (2 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^4 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{63} \sqrt{-(4+m)(-4+m)} m^2) \\
& - \frac{1}{68762925} (8 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} A \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} p^4) \\
& - \frac{1}{843679831995} (544 m^2 p^6 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& - \frac{1}{8035046019} (16 A R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} p^4 \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& - \frac{1}{124804708875} (5552 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{99} \sqrt{-(5+m)(-5+m)} m^2 p^6) \\
& + \frac{1}{1004380752375} (4 R^4 Q I^4 \sqrt{15} \sqrt{-(2+m)(-2+m)} \sqrt{35} \sqrt{-(3+m)(-3+m)} p^4 \\
& \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& - \frac{1}{63275987399625} (2 R^2 Q I^2 \sqrt{15} \sqrt{-(2+m)(-2+m)} p^8 \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{21} \sqrt{(2+m)(-2+m)(3+m)(-3+m)} \\
& \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)}) + \frac{20620 R^4 Q I^4}{143} \\
& + \frac{1613251792 m^2 p^4 R^2 Q I^2}{3468465} + \frac{602933336 m^6 p^4 Q I^2 R^2}{31216185} - \frac{91459408 m^4 p^4 R^2 Q I^2}{1486485}
\end{aligned}$$

$$\begin{aligned}
& + \frac{28838363824 p^2 A R^2 Q I^2}{10405395} + \frac{4920 A^2 Q I^2 R^2 m^2}{13} - \frac{1296000 R^2 Q I^2}{13} \\
& + \frac{237419296 m^4 p^2 Q I^2 R^2}{495495} - \frac{3476510864 m^2 p^2 R^2 Q I^2}{495495} + \frac{64368 A Q I^2 R^2 m^2}{13} \\
& + \frac{24327790148 p^2 A^2 R^2 Q I^2}{156080925} - \frac{5268491516 p^4 A R^2 Q I^2}{66891825} + \frac{8 A Q I^4 R^4 m^4}{55} \\
& - \frac{1652 A R^4 Q I^4 m^2}{429} + \frac{460 A^3 Q I^2 R^2 m^2}{39} \\
& - \frac{1}{1406133053325} \left(2 A \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right. \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 p^6 \Big) \\
& - \frac{1}{7030665266625} \left(4 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right. \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \\
& \sqrt{117} \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} Q I^2 R^2 m^2 \\
& p^8 \Big) \\
& + \frac{1}{9810801} \left(4 R^4 Q I^4 \sqrt{99} \sqrt{-(5+m)(-5+m)} p^2 \sqrt{117} \right. \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} \Big) \\
& + \frac{1}{297972675} \left(710888 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \right. \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} p^4 \Big) \\
& - \frac{1}{3443591151} \left(2168 p^6 R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \right. \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)} \Big) \\
& - \frac{1}{74882825325} \left(538388 p^6 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \right. \\
& \sqrt{-(5+m)(-5+m)} \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)} \Big) \\
& - \frac{1}{2205} \left(16 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} p^2 \sqrt{45} \right. \\
& \left. \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \right)
\end{aligned}$$

$$\begin{aligned}
& + \frac{1}{11036025} (16 R^4 Q I^4 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} p^2) \\
& - \frac{1}{13408770375} (859828 R^2 Q I^2 \sqrt{35} \sqrt{-(3+m)(-3+m)} \sqrt{45} \\
& \sqrt{(3+m)(-3+m)(4+m)(-4+m)} \sqrt{63} \sqrt{-(4+m)(-4+m)} p^6) \\
& + \frac{1}{29432403} (200 p^4 R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)}) \\
& - \frac{1}{68607} (16 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} p^2 \sqrt{77} \\
& \sqrt{(4+m)(-4+m)(5+m)(-5+m)}) \\
& + \frac{1}{237723255} (22136 R^2 Q I^2 \sqrt{63} \sqrt{-(4+m)(-4+m)} \sqrt{99} \sqrt{-(5+m)(-5+m)} p^4 \\
& \sqrt{77} \sqrt{(4+m)(-4+m)(5+m)(-5+m)}) \\
& - \frac{1}{2024451} (32 R^2 Q I^2 \sqrt{99} \sqrt{-(5+m)(-5+m)} p^2 \sqrt{117} \\
& \sqrt{(5+m)(-5+m)(6+m)(-6+m)} \sqrt{143} \sqrt{-(6+m)(-6+m)})
\end{aligned}$$

Factoring this expression fails. For the particular case of ${}^L H_2^{-+}$ with $Z_A = Z_B$ for which $q_1 = 0$, we form the corresponding matrix of order 6,

> Fmm := subs(Q1=0, Fm(6));

$$\begin{aligned}
Fmm := & \left[\left[-2 + \frac{p^2(-2m^2+3)}{5} - A, 0, \frac{p^2 \sqrt{21} \sqrt{(2+m)(2-m)(3+m)(3-m)}}{105}, 0, \right. \right. \\
& \left. \left. 0, 0 \right], \right. \\
& \left[0, -6 + \frac{p^2(-2m^2+11)}{21} - A, 0, \frac{p^2 \sqrt{45} \sqrt{(3+m)(3-m)(4+m)(4-m)}}{315}, 0, 0 \right. \\
& \left. \right], \\
& \left[\frac{p^2 \sqrt{21} \sqrt{(2+m)(2-m)(3+m)(3-m)}}{105}, 0, -12 + \frac{p^2(-2m^2+23)}{45} - A, 0, \right. \\
& \left. \frac{p^2 \sqrt{77} \sqrt{(4+m)(4-m)(5+m)(5-m)}}{693}, 0 \right],
\end{aligned} \tag{3.3}$$

$$\begin{aligned}
& \left[0, \frac{p^2 \sqrt{45} \sqrt{(3+m)(3-m)(4+m)(4-m)}}{315}, 0, -20 + \frac{p^2(-2m^2+39)}{77} - A, 0, \right. \\
& \left. \frac{p^2 \sqrt{117} \sqrt{(5+m)(5-m)(6+m)(6-m)}}{1287} \right], \\
& \left[0, 0, \frac{p^2 \sqrt{77} \sqrt{(4+m)(4-m)(5+m)(5-m)}}{693}, 0, -30 + \frac{p^2(-2m^2+59)}{117} - A, 0 \right. \\
& \left. \right], \\
& \left[0, 0, 0, \frac{p^2 \sqrt{117} \sqrt{(5+m)(5-m)(6+m)(6-m)}}{1287}, 0, -42 + \frac{p^2(-2m^2+83)}{165} \right. \\
& \left. - A \right]
\end{aligned}$$

which exhibits a tridiagonal form of which the first superdiagonal and first subdiagonal contain only zero elements. The determinant of the latter matrix is

> DFm := LinearAlgebra:-Determinant(Fmm);

$$\begin{aligned}
DFm := & 3628800 + 3110400 A - \frac{22832640}{13} p^2 + \frac{5053}{920205} p^{12} + A^6 + \frac{16}{4347968625} m^{12} p^{12} \quad (3.4) \\
& - \frac{1744}{4347968625} m^{10} p^{12} + \frac{7652}{483107625} m^8 p^{12} - \frac{1217512}{4347968625} m^6 p^{12} \\
& + \frac{9660316}{4347968625} m^4 p^{12} - \frac{9968248}{1449322875} m^2 p^{12} - \frac{283886}{2760615} A p^{10} + \frac{298877}{418275} A^2 p^8 \\
& - \frac{2276324}{975975} A^3 p^6 + \frac{24877}{6435} A^4 p^4 - \frac{614}{195} A^5 p^2 + \frac{1949155616}{869593725} m^2 p^{10} \\
& - \frac{107913616}{173918745} m^4 p^{10} + \frac{28352}{869593725} m^{10} p^{10} - \frac{2141792}{869593725} m^8 p^{10} + \frac{2574592}{41409225} m^6 p^{10} \\
& - \frac{11536}{39} p^2 A^4 + \frac{1879792}{6435} p^4 A^3 + \frac{79519168}{2927925} p^8 A - \frac{388497808}{2927925} p^6 A^2 + 112 A^5 \\
& - \frac{27054656}{13803075} p^{10} + \frac{17569304}{75075} p^8 + 4648 A^4 + \frac{32807920}{143} p^4 - \frac{6618787712}{585585} p^6 \\
& + 89280 A^3 + \frac{116}{195} A^5 m^2 p^2 + \frac{976}{869593725} A m^{10} p^{10} - \frac{1208}{13378365} A m^8 p^{10} \\
& + \frac{3044}{26351325} A^2 m^8 p^8 + \frac{101408}{41409225} A m^6 p^{10} - \frac{296}{46475} A^2 m^6 p^8 + \frac{43136}{8783775} A^3 m^6 p^6 \\
& - \frac{4612568}{173918745} A m^4 p^{10} + \frac{26516}{250965} A^2 m^4 p^8 - \frac{207712}{1254825} A^3 m^4 p^6 + \frac{91582156}{869593725} A m^2 p^{10}
\end{aligned}$$

$$\begin{aligned}
& + \frac{568}{6435} A^4 m^4 p^4 - \frac{15065168}{26351325} A^2 m^2 p^8 + \frac{1719752}{1254825} A^3 m^2 p^6 - \frac{1912}{1287} A^4 m^2 p^4 \\
& + \frac{2464}{39} m^2 p^2 A^4 + \frac{53488}{6435} m^4 p^4 A^3 + \frac{167008}{26351325} m^8 p^8 A - \frac{2855968}{8783775} m^6 p^8 A \\
& + \frac{3312544}{8783775} m^6 p^6 A^2 + \frac{1244816}{250965} m^4 p^8 A - \frac{14666768}{1254825} m^4 p^6 A^2 - \frac{162880}{1287} m^2 p^4 A^3 \\
& - \frac{641795296}{26351325} m^2 p^8 A + \frac{109981408}{1254825} m^2 p^6 A^2 + \frac{475328}{195} m^2 p^2 A^3 + \frac{8793536}{975975} m^6 p^6 A \\
& - \frac{106293856}{418275} m^4 p^6 A + \frac{1741936}{6435} m^4 p^4 A^2 + \frac{712994336}{418275} m^2 p^6 A - \frac{23661776}{6435} m^2 p^4 A^2 \\
& + \frac{4104768}{13} m^2 p^2 A + \frac{7656256}{2145} m^4 p^4 A - \frac{91177088}{2145} m^2 p^4 A + \frac{2729088}{65} m^2 p^2 A^2 \\
& - \frac{2093791376}{8783775} m^2 p^8 + \frac{720704}{8783775} m^8 p^8 - \frac{11383808}{2927925} m^6 p^8 + \frac{22617968}{418275} m^4 p^8 \\
& - \frac{1931552}{195} p^2 A^3 - \frac{601398368}{266175} p^6 A + \frac{47614768}{6435} p^4 A^2 + \frac{10460160}{13} m^2 p^2 \\
& + \frac{2265280}{143} m^4 p^4 - \frac{23134016}{143} m^2 p^4 - \frac{11541216}{13} p^2 A + 808848 A^2 + \frac{38870144}{585585} m^6 p^6 \\
& - \frac{46917056}{27885} m^4 p^6 + \frac{277175168}{27885} m^2 p^6 + \frac{156343936}{2145} p^4 A - \frac{9387072}{65} p^2 A^2
\end{aligned}$$

We simplify this determinant,

$$\begin{aligned}
& > \text{DFm} := \text{factor}(\text{subs}(m=0, \text{DFm})); \\
& DFm := \frac{1}{96621525} \left((-35 p^6 + 315 p^4 A - 693 p^2 A^2 + 4788 p^4 + 429 A^3 - 20856 p^2 A \right. \\
& \quad + 18876 A^2 - 111012 p^2 + 190476 A + 308880) (-15159 p^6 + 147455 p^4 A \\
& \quad - 345345 p^2 A^2 + 3337180 p^4 + 225225 A^3 - 15735720 p^2 A + 15315300 A^2 \\
& \quad \left. - 141441300 p^2 + 272972700 A + 1135134000) \right)
\end{aligned} \tag{3.5}$$

into two factors in the numerator and a denominator; the denominator is

$$\begin{aligned}
& > \text{DFmd} := \text{op}(1, \text{DFm}); \\
& DFmd := \frac{1}{96621525}
\end{aligned} \tag{3.6}$$

and the two factors in the numerator are

$$\begin{aligned}
& > \text{DFm1} := \text{collect}(\text{op}(2, \text{DFm}), A); \\
& DFm1 := 429 A^3 + (-693 p^2 + 18876) A^2 + (315 p^4 - 20856 p^2 + 190476) A - 35 p^6 \\
& \quad + 4788 p^4 - 111012 p^2 + 308880
\end{aligned} \tag{3.7}$$

```
> DFm1 := collect(simplify(DFm1/975), A);
```

$$DFm1 := \frac{11A^3}{25} + \left(-\frac{231p^2}{325} + \frac{484}{25} \right) A^2 + \left(\frac{21}{65} p^4 - \frac{6952}{325} p^2 + \frac{4884}{25} \right) A - \frac{7p^6}{195} \quad (3.8)$$

$$+ \frac{1596p^4}{325} - \frac{37004p^2}{325} + \frac{1584}{5}$$

and

```
> DFm2 := collect(op(3, DFm), A);
```

$$DFm2 := 225225A^3 + (-345345p^2 + 15315300)A^2 + (147455p^4 - 15735720p^2 + 272972700)A - 15159p^6 + 3337180p^4 - 141441300p^2 + 1135134000 \quad (3.9)$$

If $q_1 = 0$, i.e. for symmetric atomic nuclei, the above pentadiagonal matrix divides into tridiagonal matrices, even, of which for order 3 the determinant is formed with this procedure,

```
> dFe := proc(k)
```

```
  if k = 0 or type(k, posint) then
```

```
    if k = 0 then
```

```
      1;
```

```
    elif k = 1 then
```

```
      1/3*x-A;
```

```
    elif 1 < k then
```

```
      (-2*(k-1)*(2*k-1)+x*(8*(k-1)^2-5+4*k)/(4*k-1)/(4*k-5)-A)*dFe
      (k-1)-4*x^2/(4*k-5)^2
```

```
      *(2*k-3)^2*(k-1)^2/(4*k-7)/(4*k-3)*dFe(k-2);
```

```
    end if;
```

```
  else
```

```
    ('dFe')(k);
```

```
  end if;
```

```
end proc;
```

```
> collect(subs(x=p^2, numer(dFe(3))), A);
```

$$-231A^3 + (315p^2 - 6006)A^2 + (-105p^4 + 5124p^2 - 27720)A + 5p^6 - 630p^4 + 9240p^2 \quad (3.10)$$

which expression is the negative of a factor above of the determinant of the tridiagonal matrix, and odd, of which from the matrix of order 3 the determinant is formed with this procedure,

```
> dFo := proc(k)
```

```
  if k = 0 or type(k, posint) then
```

```
    if k = 0 then
```

```
      1;
```

```
    elif k = 1 then
```

```
      -2+3/5*x-A;
```

```
    elif 1 < k then
```

```
      (-2*(2*k-1)*k+x*(2*(2*k-1)^2-3+4*k)/(4*k+1)/(4*k-3)-A)*dFo
      (k-1)-4*x^2/(4*k-3)^2*(k-1)^2
```

```

        *(2*k-1)^2/(4*k-5)/(4*k-1)*dFo(k-2);
    end if;
else
    'dFo'(k);
end if;
end proc;
> collect(subs(x=p^2, numer(dFo(3))), A);
-429 A^3 + (693 p^2 - 18876) A^2 + (-315 p^4 + 20856 p^2 - 190476) A + 35 p^6 - 4788 p^4
+ 111012 p^2 - 308880

```

(3.11)

which expression is again the negative of a factor above of the determinant of the tridiagonal matrix. For the ξ coordinate, we use a basis of Hylleraas functions, hence in terms of Laguerre polynomials,

$$\Xi(\xi) = e^{-p(\xi-1)} (2p(\xi-1))^{\frac{m}{2}} \sum_{n=\frac{m}{2}} C_{n-\frac{m}{2}, m} L_{n-\frac{m}{2}, m}(2p(\xi-1))$$

which leads to matrices **B** and **Q** formed in the following procedures, with tests to show their structure.

```

> Qm := proc(N)
    local i, Q;
    global p, A, m, Q2, R;
    Q := Matrix(N, N, shape=symmetric);
    for i to N do
        Q[i, i] := (2*i+1)*(R*Q2/(2*p)-i-1-2*p)+m^2/4+i+R*Q2-p^2+A;
    end do;
    for i to N-1 do
        Q[i+1, i] := -((i+m/2+1)*(i-m/2+1))^(1/2)*(R*Q2/(2*p)-i-1);
    end do;
    Q;
end proc;
> Qm(3);

```

$$\left[\left[\frac{3 R Q2}{2 p} - 5 - 6 p + \frac{m^2}{4} + R Q2 - p^2 + A, -\sqrt{\left(2 + \frac{m}{2}\right) \left(2 - \frac{m}{2}\right)} \left(\frac{R Q2}{2 p} - 2\right), 0 \right. \right. \quad (3.12)$$

$$\left. \right],$$

$$\left[-\sqrt{\left(2 + \frac{m}{2}\right) \left(2 - \frac{m}{2}\right)} \left(\frac{R Q2}{2 p} - 2\right), \frac{5 R Q2}{2 p} - 13 - 10 p + \frac{m^2}{4} + R Q2 - p^2 \right.$$

$$\left. + A, -\sqrt{\left(3 + \frac{m}{2}\right) \left(3 - \frac{m}{2}\right)} \left(\frac{R Q2}{2 p} - 3\right) \right],$$

$$\left[0, -\sqrt{\left(3 + \frac{m}{2}\right) \left(3 - \frac{m}{2}\right)} \left(\frac{R Q2}{2 p} - 3\right), \frac{7 R Q2}{2 p} - 25 - 14 p + \frac{m^2}{4} + R Q2 - p^2 \right]$$

$$+ A \Bigg]$$

> LinearAlgebra:-Determinant(%);

$$\begin{aligned} & \frac{1}{64 p^3} (8 Q^2 R^3 m^2 p + 64 Q^2 R^3 p^3 + 2 Q^2 R^2 m^4 p + 40 Q^2 R^2 m^2 p^3 - 192 Q^2 R^2 p^5 \\ & + 12 Q^2 R m^4 p^3 - 96 Q^2 R m^2 p^5 + 192 Q^2 R p^7 + m^6 p^3 - 12 m^4 p^5 + 48 m^2 p^7 - 64 p^9 \\ & + 8 A Q^2 R^2 m^2 p + 192 A Q^2 R^2 p^3 + 96 A Q^2 R m^2 p^3 - 384 A Q^2 R p^5 + 12 A m^4 p^3 \\ & - 96 A m^2 p^5 + 192 A p^7 + 20 Q^2 R^3 m^2 + 480 Q^2 R^3 p^2 + 80 Q^2 R^2 m^2 p^2 \\ & - 2880 Q^2 R^2 p^4 + 10 Q^2 R m^4 p^2 - 1120 Q^2 R m^2 p^4 + 4320 Q^2 R p^6 - 120 m^4 p^4 \\ & + 960 m^2 p^6 - 1920 p^8 + 192 A^2 Q^2 R p^3 + 48 A^2 m^2 p^3 - 192 A^2 p^5 + 960 A Q^2 R^2 p^2 \\ & + 160 A Q^2 R m^2 p^2 - 4800 A Q^2 R p^4 - 960 m^2 p^4 A + 3840 p^6 A + 928 Q^2 R^3 p \\ & - 72 Q^2 R^2 m^2 p - 12768 Q^2 R^2 p^3 - 2704 Q^2 R m^2 p^3 + 32768 Q^2 R p^5 - 120 m^4 p^3 \\ & + 5712 m^2 p^5 - 20928 p^7 + 64 A^3 p^3 + 480 A^2 Q^2 R p^2 - 1920 p^4 A^2 + 928 A Q^2 R^2 p \\ & - 14592 A Q^2 R p^3 - 1168 A m^2 p^3 + 23680 A p^5 + 400 Q^2 R^3 - 18560 Q^2 R^2 p^2 \\ & - 1080 Q^2 R m^2 p^2 + 100480 Q^2 R p^4 + 10720 m^2 p^4 - 103680 p^6 - 2752 A^2 p^3 \\ & - 10240 A Q^2 R p^2 + 49920 p^4 A - 6464 Q^2 R^2 p + 114560 Q^2 R p^3 + 4368 m^2 p^3 \\ & - 237440 p^5 + 26752 A p^3 + 32960 Q^2 R p^2 - 217600 p^4 - 52480 p^3) \end{aligned} \quad (3.13)$$

> collect(subs(m=0, %), A, factor);

$$\begin{aligned} & A^3 + \frac{(6 Q^2 R p - 6 p^3 + 15 R Q^2 - 60 p^2 - 86 p) A^2}{2 p} + \frac{1}{2 p^2} ((6 Q^2 R^2 p^2 - 12 Q^2 R p^4 \\ & + 6 p^6 + 30 Q^2 R^2 p - 150 Q^2 R p^3 + 120 p^5 + 29 Q^2 R^2 - 456 Q^2 R p^2 + 740 p^4 \\ & - 320 Q^2 R p + 1560 p^3 + 836 p^2) A) + \frac{1}{4 p^3} (4 Q^2 R^3 p^3 - 12 Q^2 R^2 p^5 \\ & + 12 Q^2 R p^7 - 4 p^9 + 30 Q^2 R^3 p^2 - 180 Q^2 R^2 p^4 + 270 Q^2 R p^6 - 120 p^8 + 58 Q^2 R^3 p \\ & - 798 Q^2 R^2 p^3 + 2048 Q^2 R p^5 - 1308 p^7 + 25 Q^2 R^3 - 1160 Q^2 R^2 p^2 + 6280 Q^2 R p^4 \\ & - 6480 p^6 - 404 Q^2 R^2 p + 7160 Q^2 R p^3 - 14840 p^5 + 2060 Q^2 R p^2 - 13600 p^4 \\ & - 3280 p^3) \end{aligned} \quad (3.14)$$

> Bm := proc(N)

local i,B;

global p,A,m,Q2,R;

B := Matrix(N,N, shape=symmetric);


```

for i to N do
  B[i,i] := 4*p+2*i+1;
end do;
for i to N-1 do
  B[i+1,i] := -((i+m/2+1)*(i-m/2+1))^(1/2);
end do;
B;
end proc:
> Bm(3);

```

$$\begin{bmatrix} 4p+3 & -\sqrt{\left(2+\frac{m}{2}\right)\left(2-\frac{m}{2}\right)} & 0 \\ -\sqrt{\left(2+\frac{m}{2}\right)\left(2-\frac{m}{2}\right)} & 4p+5 & -\sqrt{\left(3+\frac{m}{2}\right)\left(3-\frac{m}{2}\right)} \\ 0 & -\sqrt{\left(3+\frac{m}{2}\right)\left(3-\frac{m}{2}\right)} & 4p+7 \end{bmatrix} \quad (3.15)$$

```

> LinearAlgebra:-Determinant(%);

```

$$64p^3 + 240p^2 + 232p + 2pm^2 + 50 + \frac{5}{2}m^2 \quad (3.16)$$

```

> collect(%, p, factor);

```

$$64p^3 + 240p^2 + (2m^2 + 232)p + 50 + \frac{5m^2}{2} \quad (3.17)$$

The determinants of two additional matrices are formed in the following procedures.

```

> dR := proc(k)
  if k = 0 or type(k, posint) then
    if k = 0 then
      1;
    elif k = 1 then
      R/p-1-2*p+2*R-p^2+A;
    elif 1 < k then
      ((2*k-1)*(R/p-k-2*p)+k-1+2*R-p^2+A)*dR(k-1)-(k-1)^2*(R/p-
      k+1)^2*dR(k-2);
    end if;
  else
    ('dR')(k);
  end if;
end proc:
> dR(3);

```

$$\left(\frac{5R}{p} - 13 - 10p + 2R - p^2 + A\right) \left(\left(\frac{3R}{p} - 5 - 6p + 2R - p^2 + A\right) \left(\frac{R}{p} - 1 - 2p\right) \right. \quad (3.18)$$

$$+ 2R - p^2 + A) - \left(\frac{R}{p} - 1\right)^2 - 4\left(\frac{R}{p} - 2\right)^2 \left(\frac{R}{p} - 1 - 2p + 2R - p^2 + A\right)$$

```
> dY := proc(k)
    if m = 0 then
        dR(k);
    end if;
end proc:
> dY(3);
```

We apply these matrices in sections 3.64 and 3.65 to calculate, for $^1\text{H}_2^{+}$ in its electronic ground state Σ_g^{+} , the electronic energy as a function of internuclear distance, and the binding energy and equilibrium internuclear distance.

3.64 $V(R)$ for H_2^{+}

```
> # no restart here
```

For this calculation to evaluate the total electronic energy as a function of internuclear separation R , we form the resultant of those two determinants dR and dFe .

```
> V := proc(N, Rr)
    local dRN, dFeN, RAres, RNp, F, sv, pp, j;
    dRN := dR(N);
    dFeN := subs(x = p^2, dFe(N));
    RAres := resultant( numer(dRN), numer(dFeN), A);
    RNp := subs(R = Rr, RAres);
    F := [fsolve(RNp=0, {p}, complex)];
    sv := NULL;
    for j in F do
        pp := subs(j, p);
        if type(pp, numeric) and pp>0 then
            sv := sv, pp;
        end if;
    end do;
    sv := max(sv);
    -2*sv^2/Rr^2 + 1/Rr;
end proc:
> Digits := 10;
```

Digits := 10

(4.1)

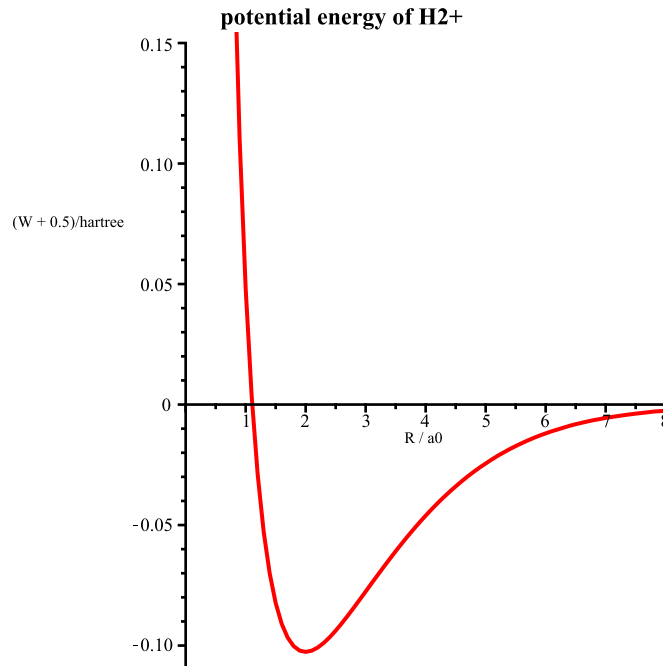
```
> J := 80:
N := 5:
for j from 7 to J do
    Rr[j] := Float(j, -1);
    Et[j] := V(N, Rr[j]);
```

```
end do:
```

```
lp[N] := [seq([Rr[j],Et[j]+0.5], j=7..J)]:
```

We plot these data as the total energy that becomes the potential energy $V(R)$ of ${}^1\text{H}_2^{+}$ for nuclear vibrations in electronic ground state ${}^2\Sigma_g^{+}$, with atomic unit a_0 for internuclear distance R and unit hartree for energy; the energy scale is relative to $W = 0$ for separation into ${}^1\text{H}$ and ${}^1\text{H}^{+}$.

```
> plot([seq(lp[j], j=5..5)], labels=["R / a0", "(W + 0.5)/hartree"],
       colour=red,
       title="potential energy of H2+", titlefont=[TIMES,BOLD,14],
       view=[0..8,-0.11..0.15]);
```



This curve exhibits a minimum of energy near internuclear separation $R = 2 a_0$ that we define accurately in section 3.65.

3.65 equilibrium internuclear separation and energy for H_2^{+}

```
> # no restart here
```

To evaluate the equilibrium internuclear separation and the corresponding electronic energy, we invoke procedures **dR** and **dFe** in section 3.63 within the following procedures to solve five simultaneous non-linear equations for $R, A, p, \frac{\partial}{\partial R} A$ and $\frac{\partial}{\partial R} p$.

```
> Digits := 50;
```

```
Digits := 50
```

(5.1)

```
> H2p1 := proc()
```

```
    local dFe2,dR2,exs,si,bi,i,Eelec,Etot,dEtot,eqs;
    global p,x;
```

```

dFe2 := expand(subs(x=p^2, dFe(1)));
dR2 := expand(dR(1));
Eelec := -2*p^2/R^2;
Etot := Eelec + 1/R;
dEtot := diff(Etot,R) + diff(Etot,p)*dpdR;
eqs := {dEtot = 0, dFe2 = 0, diff(dFe2,p)*dpdR + diff(dFe2,A)*
dAdR = 0,
        dR2 = 0, diff(dR2,p)*dpdR + diff(dR2,A)*dAdR +diff(dR2,
R) = 0};
exs := solve(eqs);
exs := evalf([allvalues(exs)]);
for i to nops(exs) do
    si := exs[i];
    bi := map(rhs, si);
    bi := map(type, bi, positive);
    if bi = {true} then
        break;
    end if;
end do;
si;
end proc:
> initsol1 := H2p1():
> H2pmin := proc(N, initsol)
    local dFe2,Eelec,Etot,dEtot,eqs,dR2;
    Eelec := -2*p^2/R^2;
    Etot := Eelec + 1/R;
    dEtot := diff(Etot,R) + diff(Etot,p)*dpdR;
    dFe2 := expand(subs(x=p^2, dFe(N)));
    dR2 := expand(dR(N));
    eqs := {dEtot = 0, dFe2 = 0, diff(dFe2,p)*dpdR + diff(dFe2,A)*
dAdR = 0,
        dR2 = 0, diff(dR2,p)*dpdR + diff(dR2,A)*dAdR +diff(dR2,
R) = 0};
    fsolve(eqs, initsol);
end proc:
> Eelec := -2*p^2/R^2;
    Etot := Eelec + 1/R;

```

$$E_{elec} := -\frac{2p^2}{R^2}$$

$$E_{tot} := -\frac{2p^2}{R^2} + \frac{1}{R}$$

(5.2)

```

> N := 22:
> sv := Vector(N):
> sv[1] := initsol1:
> for j from 2 to N do
    sv[j] := H2pmin(j, sv[j-1]);
end do:

```

In the following table, N denotes the order of the matrix applied to evaluate the equilibrium internuclear distance, and E_{\min} denotes the energy for electronic ground state $^2\Sigma_g^+$ at that distance relative to zero for H and H $^+$ at infinite internuclear separation.

```

> print('N', ``, 'Re `'/a[0]', ``, 'E[min]/hartree', ``, ``);

```

```

for i from 1 to N do
    svi := subs(sv[i], Etot+0.5):
    print(i, evalf(subs(sv[i], R),16), evalf(svi,16));
end do:

```

N ,	$\frac{Re}{a_0}$,	$\frac{E_{\min}}{\text{hartree}}$,
1,	1.847210745982953,	−0.07601441482706901
2,	1.996299537003018,	−0.1024988653239447
3,	1.997241013166768,	−0.1026316234217334
4,	1.997197909857719,	−0.1026344040069634
5,	1.997193870758813,	−0.1026345976364212
6,	1.997193399194489,	−0.1026346164243632
7,	1.997193333100135,	−0.1026346187098893
8,	1.997193322410121,	−0.1026346190395379
9,	1.997193320468660,	−0.1026346190939259
10,	1.997193320080437,	−0.1026346191039399
11,	1.997193319996208,	−0.1026346191059609
12,	1.997193319976603,	−0.1026346191064020
13,	1.997193319971750,	−0.1026346191065051
14,	1.997193319970483,	−0.1026346191065306
15,	1.997193319970135,	−0.1026346191065373
16,	1.997193319970035,	−0.1026346191065391
17,	1.997193319970006,	−0.1026346191065397

18, 1.997193319969997, -0.1026346191065398
 19, 1.997193319969994, -0.1026346191065399
 20, 1.997193319969993, -0.1026346191065399
 21, 1.997193319969992, -0.1026346191065399
 22, 1.997193319969992, -0.1026346191065399 (5.3)

This table shows that the equilibrium internuclear distance converges to 16 significant digits within order 22 of matrices **F** and **R**.

That curve of electronic energy versus internuclear distance might serve as a potential energy for the motion of the atomic nuclei, yielding discrete states of particular energies, just as the parabolic potential energy of the canonical linear harmonic oscillator yields energies $E_n = \left(n + \frac{1}{2}\right) h \nu_0$. The energies of the states of ${}^1\text{H}_2^+$ have, however, not a constant interval, and the calculation of those energies has no exact algebraic solution; for that reason we omit here its consideration, but numerical solutions appear elsewhere. The energies of ${}^1\text{H}_2^+$ in its discrete states supported by that function of potential energy tend to converge toward the molecular dissociation limit that produces ${}^1\text{H} + {}^1\text{H}^+$, similarly as the energies of the electronic states of ${}^1\text{H}$ tend to converge towards the atomic ionization limit that produces ${}^1\text{H}^+ + e^-$, with a continuum of energies above those limits in both cases.

e3.61 exercise

Apply the result, in section 3.65, of the dependence of total energy of H_2^+ as a function of internuclear distance R to calculate the energies of the some vibrational and some rotational states using procedures in sections 1.45 and 1.46.

3.66 behaviour of energy of H_2^+ toward limits of united and separate atoms

> **restart:**

These calculations [B. Nickel, J Phys A Math Theor 44, 395301 (2011)] of the energy, in atomic unit hartree throughout and based on solutions of Schroedinger's equation as continued fractions, treat dihydrogen molecular cation ${}^1\text{H}_2^+$ in electronic ground state ${}^2\Sigma_g^+$ in the region of molecular binding and approaching the limits of the united and separate atoms. We execute this loop twice, here with **Digits=25**, and below with increased precision.

```
> E1 := []:
E2 := []:
E3 := []:
for N from 1 to 150 do
    # Set R (atomic units)
    R := N/20:
    if N>100 then
        R := (N - 90)/2;
    end if:
    # Set continued fraction lengths Na, Nr
```

```

Digits := 25:
Na := trunc(6 + R/2):
Nr := trunc(20 + 40/R):
chk := 1:
# Execute this procedure first with the following line commented,
#   and then with the comment character removed.
# Digits := 30: Na := trunc(8+R/2): Nr := trunc(30+50/R): chk
:= 2:

      # Set initial A, k guess
A := evalf(R^2/2 + R/(1+R^2/10) - R)/2:
k := R*(1 + R/2)/(1 + R):
for loop to 10 do
      # 'angular' continued fraction and derivatives
g := 0:
dgA := 0:
dgk := 0:
for n from Na by -1 to 1 do
      an := 4*n^2*(2*n-1)^2/(4*n-3)/(4*n-1)^2/(4*n+1)*k^4:
      bn := A + 2*n*(2*n+1) - (8*n^2+4*n-1)/(4*n-1)/(4*n+3)*k^2:
      dgA := an/(bn-g)^2*(dgA-1):
      dgk := an/(bn-g)^2*(4/k*(bn-g) + dgk + 2*(8*n^2+4*n-1)/(4*
n-1)/(4*n+3)*k):
      g := an/(bn - g)
end do:
dgA := 1 - dgA:
dgk := -2*k/3 - dgk:
g := A - k^2/3 - g:
      # 'radial' continued fraction and derivatives
s := R/k - 1:
f := 2:
dfA := 0:
dfk := 0:
for n from Nr by -1 to 1 do
      an := (1 - s/n)^2:
      bn := 2 + 2*(2*k-s)/n + (k^2-(1+2*k)*s-A)/n^2:
      dfA := -1/n^2 + an/f^2*dfA:
      dfk := (2+R/k^2)/n*(2+1/n) + 2*k/n^2 - 2/n*(1-s/n)*R/k^2/f +
an/f^2*dfk:
      f := bn - an/f
end do:
dfA := 1 - s^2/f^2*dfA:
dfk := -2 - 2*k - R/k^2*(1+2*s/f) - s^2/f^2*dfk:

```

```

f := A + (1+2*k)*s - k^2 + s^2/f:
      # Newton-Raphson update of A, k
det := dfA*dgk - dgA*dfk:
A := A - (dgk*f - dfk*g)/det:
k := k - (dfA*g - dgA*f)/det:
if evalf(log10(abs(f)+abs(g))) < 5-Digits then
  break;
end if;
end do:

      # output (R, E in atomic units)
E1 := [op(E1), [R, -2*k^2/R^2]]:      # electron
energy only
E2 := [op(E2), [R, 1/R+1/2-2*k^2/R^2]]:  # p-p repulsion
included
E3 := [op(E3), [R, R^4*(1/R+1/2-2*k^2/R^2)]]:  # with R^4
scaling
`print(loop, g, f)`:
end do:

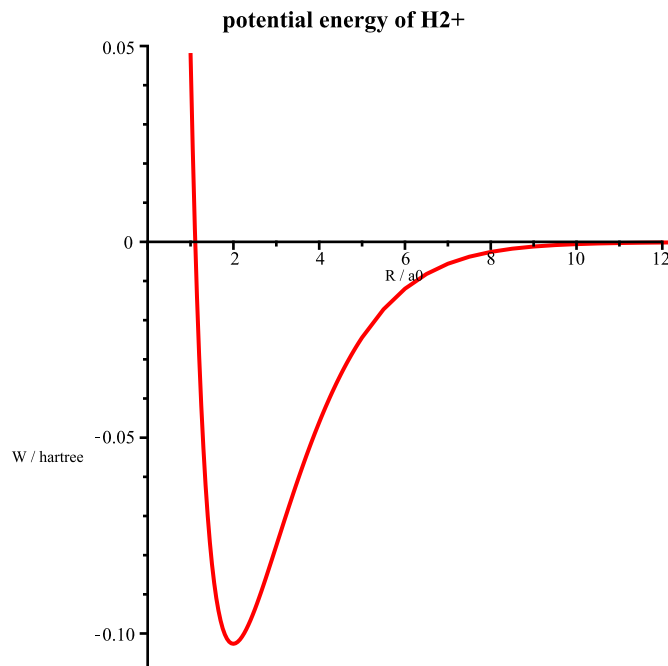
```

We show first the exact total energy/hartree of ${}^1\text{H}_2^+$ as a function of internuclear separation R /bohr over the domain of molecular binding, relative to $E=0$ for the separate products ${}^1\text{H}$ and ${}^1\text{H}^+$ of dissociation; this plot is the same as the one in section 3.65, but obtained by a separate method.

```

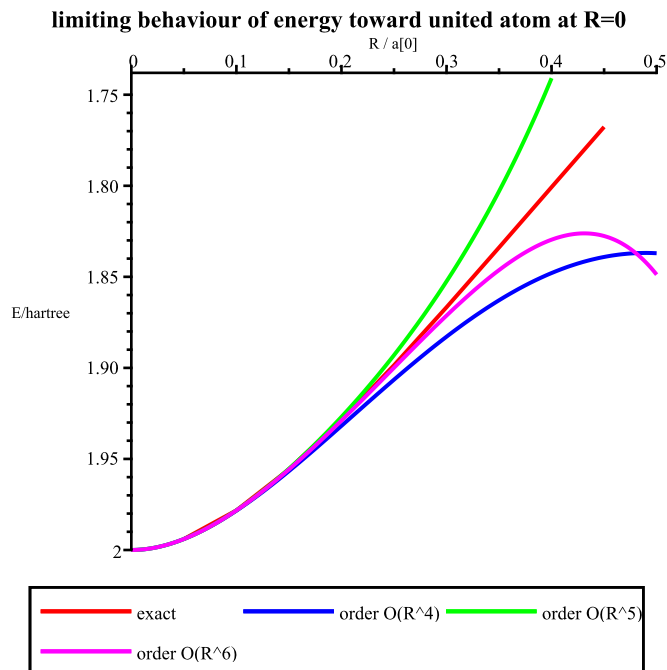
> plot(E2[20..120], view=[0..12, -0.11..0.05],
  labels=["R / a0", "W / hartree"],
  title="potential energy of H2+", titlefont=[TIMES,BOLD,14],
  colour=red );

```



We show next the limiting behaviour of the electronic energy as $R \rightarrow 0$ for the united atom, ${}^2\text{He}^{++}$, according to the exact curve -- black, and with approximations with series to $O(R^4)$ -- green, $O(R^5)$ -- blue and $O(R^6)$ -- red, all excluding the repulsion between the nuclei and relative to $\frac{1}{2}$ at the limit of separate entities ${}^1\text{H}$ and ${}^1\text{H}^{++}$. For two protons, ${}^1_1\text{H}$, in ${}^1\text{H}_2^{++}$, the united atom, ${}^2_2\text{He}$ is unstable, but for two deuterons, ${}^2_1\text{H}$, the united atom ${}^4_2\text{He}$ is stable. The electronic energy of an atom, in its electronic ground state, with one electron is $-\frac{1}{2} Z^2$ hartree, so $-\frac{1}{2}$ hartree for H or -2 hartree for He $^{++}$. The electronic energy of H_2^{++} thus decreases to a limiting value -2 hartree as $R \rightarrow 0$.

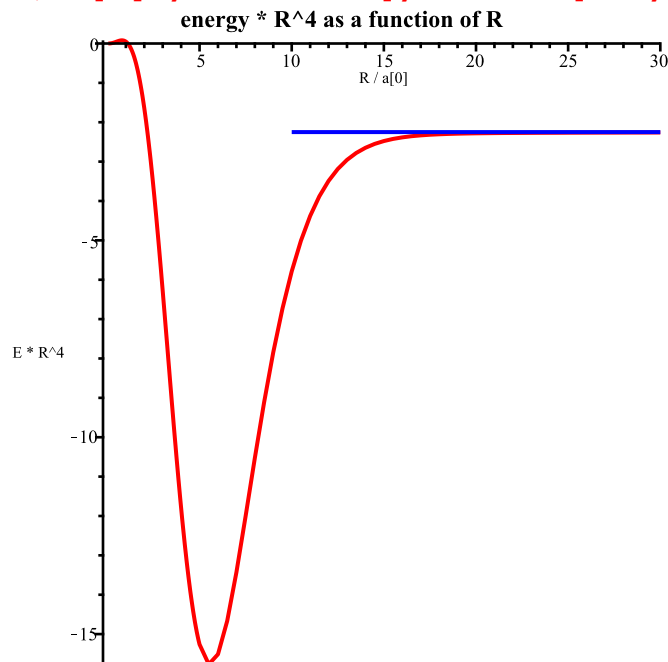
```
> Es_4 := -2+8/3*r^2 - 16/3*r^3 + 352/135*r^4:
Es_5 := Es_4 - 64/9*(ln(4*r) + gamma - 151/60)*r^5:
Es_6 := Es_5 + 128/9*(ln(4*r) + gamma - 9713/3780)*r^6:
plot([E1[1..9], [r,Es_4,r=0..0.5], [r,Es_5,r=0..0.4], [r,Es_6,r=0.
.0.5]],
      colour=[black,green,blue,red], legend=["exact", "order O(R^4)
",
      "order O(R^5)", "order O(R^6)"], titlefont=[TIMES,BOLD,14],
      title="limiting behaviour of energy toward united atom at R=
0",
      labels=["R / a[0]", "E/hartree"], colour=[red,blue,green,
magenta]));
```



The next plot of R^4 times the electronic energy shows the limiting behaviour as $R \rightarrow \infty$, namely that the curve approaches asymptotically the value $\frac{9}{4}$ that corresponds to the energy of a polarizable H atom

under the influence of a proton at distance R . In atomic units, polarizability α of H is $\frac{9}{2}$; the electric dipolar moment is thus $\underline{p} = \frac{9}{2} \underline{E}$, in which \underline{E} denotes the strength of the electric field. The energy of a polarizable atom is $-\frac{1}{2} \underline{p} \cdot \underline{E}$ or $-\frac{9}{4} E^2$. As the electric field due to a proton at distance R varies as $\frac{1}{R^2}$, the energy becomes $-\frac{9}{4} R^{-4}$, as shown by the green line in this plot. Because of the presence of factor R^4 , product $R^4 E$ approaches 0 as $R \rightarrow 0$.

```
> plot([E3, [x,-9/4, x=10..30]], titlefont=[TIMES,BOLD,14],
       title="energy * R^4 as a function of R",
       labels=["R / a[0]", "E * R^4"], colour=[red,blue]);
```



We execute the following commands,

```
> if chk=1 then
    chk1 := E1;
end if:
if chk=2 then
    chk2 := E1;
end if:
```

before repeating the calculations in the same loop as above but with the **Digits=30**.

```
> E1 := [];
E2 := [];
E3 := [];
for N from 1 to 150 do
    # Set R (atomic units)
```

```

R := N/20:
if N>100 then
  R := (N - 90)/2;
end if:

      # Set continued fraction lengths Na, Nr
Digits := 25:
Na := trunc(6 + R/2):
Nr := trunc(20 + 40/R):
chk := 1:
# Execute this procedure first with the following line commented,
#   and then with the comment character removed.
Digits := 30: Na := trunc(8+R/2): Nr := trunc(30+50/R): chk :=
2:

      # Set initial A, k guess
A := evalf(R^2/2 + R/(1+R^2/10) - R)/2:
k := R*(1 + R/2)/(1 + R):
for loop to 10 do
  # 'angular' continued fraction and derivatives
  g := 0:
  dgA := 0:
  dgk := 0:
  for n from Na by -1 to 1 do
    an := 4*n^2*(2*n-1)^2/(4*n-3)/(4*n-1)^2/(4*n+1)*k^4:
    bn := A + 2*n*(2*n+1) - (8*n^2+4*n-1)/(4*n-1)/(4*n+3)*k^2:
    dgA := an/(bn-g)^2*(dgA-1):
    dgk := an/(bn-g)^2*(4/k*(bn-g) + dgk + 2*(8*n^2+4*n-1)/(4*
n-1)/(4*n+3)*k):
    g := an/(bn - g)
  end do:
  dgA := 1 - dgA:
  dgk := -2*k/3 - dgk:
  g := A - k^2/3 - g:
  # 'radial' continued fraction and derivatives
  s := R/k - 1:
  f := 2:
  dfA := 0:
  dfk := 0:
  for n from Nr by -1 to 1 do
    an := (1 - s/n)^2:
    bn := 2 + 2*(2*k-s)/n + (k^2-(1+2*k)*s-A)/n^2:
    dfA := -1/n^2 + an/f^2*dfA:
    dfk := (2+R/k^2)/n*(2+1/n) + 2*k/n^2 - 2/n*(1-s/n)*R/k^2/f +

```

```

an/f^2*dfk:
    f := bn - an/f
end do:
dfA := 1 - s^2/f^2*dfA:
dfk := -2 - 2*k - R/k^2*(1+2*s/f) - s^2/f^2*dfk:
f := A + (1+2*k)*s - k^2 + s^2/f:
    # Newton-Raphson update of A, k
det := dfA*dgk - dgA*dfk:
A := A - (dgk*f - dfk*g)/det:
k := k - (dfA*g - dgA*f)/det:
if evalf(log10(abs(f)+abs(g))) < 5-Digits then
    break;
end if;
end do:
    # output (R, E in atomic units)
E1 := [op(E1), [R, -2*k^2/R^2]]:           # electron
energy only
E2 := [op(E2), [R, 1/R+1/2-2*k^2/R^2]]:     # p-p repulsion
included
E3 := [op(E3), [R, R^4*(1/R+1/2-2*k^2/R^2)]]: # with R^4
scaling
`print(loop, g, f)`:
end do:
if chk=1 then
    chk1 := E1;
end if:
if chk=2 then
    chk2 := E1;
end if:

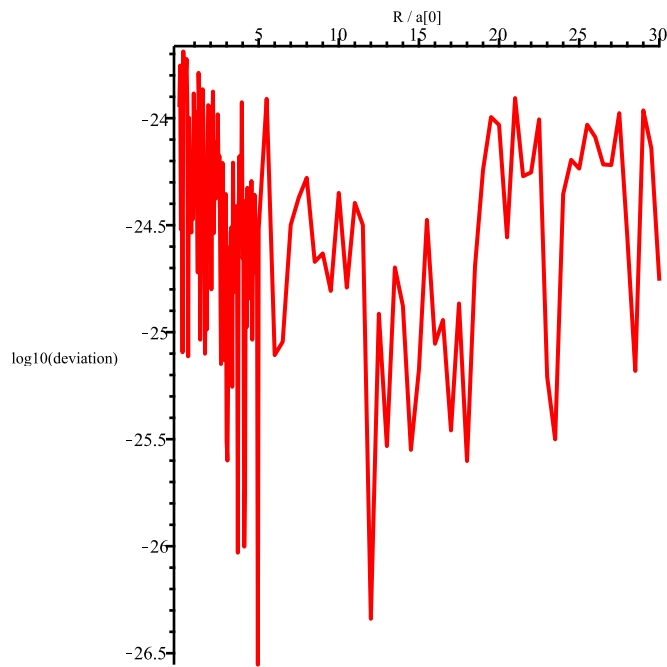
```

After the second execution of that loop with increased precision, this plot,

```

> dev := []:
for n to nops(chk2) do
    dev := [op(dev), [chk1[n][1], log10(abs(chk1[n][2] - chk2[n][2])
)]];
end do:
plot(dev, labels=["R / a[0]", "log10(deviation)"], colour=red);

```



shows that, with precision **Digits := 25**, the errors in the energies, as a function of internuclear distance, are of order 10^{-24} hartree.

These calculations on H and H_2^{+} demonstrate that great precision and even accuracy is practicable on simple atomic and molecular systems through the application of algorithms of quantum mechanics.

3.67 H_2 in singlet and triplet states

> restart;

In preceding sections we solve exactly the energies and other properties of the H atom, within the non-relativistic wave mechanics of Schrodinger, and the energy of H_2^{+} as a function of internuclear separation exactly within the non-relativistic wave mechanics with an added assumption of separation of nuclear and electronic motions; in each case the system on which the calculation is made becomes converted to the effective motion of a single particle in a field of potential energy. For H_2 that comprises two atomic nuclei and two electrons -- just as for He that comprises one atomic nucleus and two electrons, such a conversion is no longer practicable, and any illusion of exactness of a solution must be abandoned. We nevertheless proceed with an unavoidably approximate calculation, adapted from Professor M. Horbatsch with permission.

As nuclei A and B -- each a proton -- are indistinguishable, permutation symmetry of the amplitude function is applicable under the operation of exchange of those nuclei. A product model for the wavefunction

$$\Phi(r_1, r_2) = \phi_A(r_1) \phi_B(r_2)$$

yields the same energy as this wavefunction,

$$\Phi(r_1, r_2) = \phi_A(r_2) \phi_B(r_1)$$

which is hence energetically degenerate in the hamiltonian of zero order that contains the terms for only atoms A and B. We must thus consider a linear combination as a possible solution with the same eigenvalue in order zero. We choose the symmetric and antisymmetric combinations of the spatial

amplitude function as they form an overall antisymmetric state when combined with an antisymmetric spin function in the first case (singlet state, $S=0$, antiparallel, para-hydrogen) and with a symmetric spin function (three projections for $S=1$) in the second case (triplet state, parallel spins, ortho-hydrogen). The residual interactions between electron 1 and nucleus B, electron 2 and nucleus A, and between both electrons, when added destroy the degeneracy: two disparate energy levels emerge as a function of internuclear separation R . As before, we use atomic units: $\frac{h}{2\pi} = m_e = e = 1$, and distances are measured in terms of the Bohr radius. Within this frame the wavefunction of H in its electronic ground state is

```
> psi := r -> exp(-r)/sqrt(Pi);
```

$$\psi := r \mapsto \frac{e^{-r}}{\sqrt{\pi}} \quad (8.1)$$

```
> Int(subs(I=-I,psi(r))*psi(r)*r^2, r=0..infinity) = int(subs(I=-I,
psi(r))*psi(r)*r^2, r=0..infinity);
```

$$\int_0^{\infty} \frac{(e^{-r})^2 r^2}{\pi} dr = \frac{1}{4\pi} \quad (8.2)$$

To begin the calculation, for the two possible products of amplitude functions of two particles

$$\Phi_1 = \phi_A(r_1) \phi_B(r_2) \quad \text{and} \quad \Phi_2 = \phi_A(r_2) \phi_B(r_1)$$

the overlap integral is

$$S^2 = \iint \Phi_1(r_1, r_2) \Phi_2(r_1, r_2) d\tau_1 d\tau_2$$

in which τ_1 and τ_2 denote the volume elements pertaining to r_1 and r_2 respectively, and which is a function of internuclear separation R , and there is axial symmetry with respect to axis z . To treat the character of this integral in three spatial dimensions, which is a square of

$$S = \int \phi_A(r_1) \phi_B(r_2) d\tau_1$$

we use special coordinates as follows: azimuthal angle ϕ around axis z , and an orthogonal couple of distance coordinates, ξ and η , of which the lines of constant value describe ellipses and hyperbolae with the nuclei (protons) at the focal points. Internuclear separation R appears as a parameter. The relations

with cartesian coordinates follow in terms of $Rh = \frac{R}{2}$:

$$x = Rh \sqrt{(1 - \eta^2)(\xi^2 - 1)} \cos(\phi), \quad y = Rh \sqrt{(1 - \eta^2)(\xi^2 - 1)} \sin(\phi), \quad z = Rh \xi \eta$$

The radius vectors in plane xy to a point separated from nuclei A and B by distances r_A and r_B are respectively

$$r_A = Rh(\xi + \eta), \quad r_B = Rh(\xi - \eta), \quad r_A + r_B = R\xi$$

The domains of these variables are $1 \leq \xi < \infty$ which governs the size of ellipses, $-1 \leq \eta \leq +1$ for hyperbolae, and $0 \leq \phi \leq 2\pi$. The volume element is $d\tau = Rh^3 (\xi^2 - \eta^2) d\xi d\eta d\phi$. Overlap matrix element S is hence an integral over

$$\psi(r_A) \psi(r_B) = \psi(r_A + r_B).$$

```
> assume(Rh > 0);
```

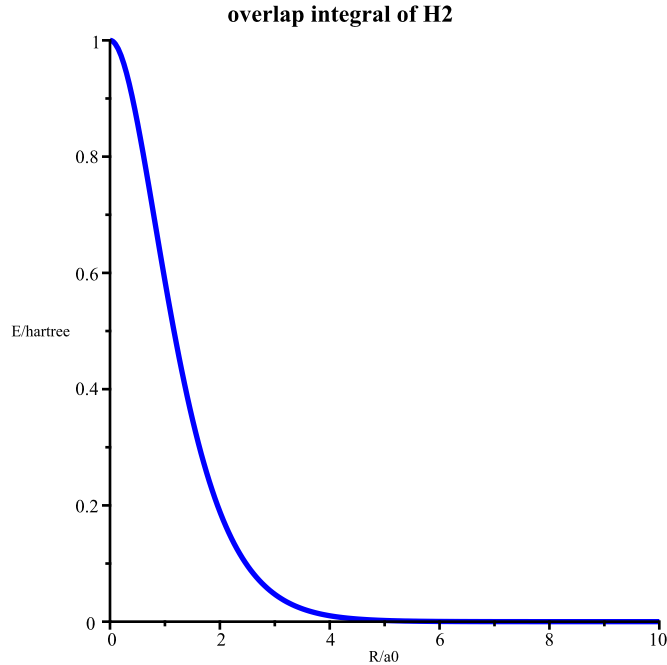
```
> S := 2*Pi*Rh^3*int(int(psi(Rh*(xi+eta))*psi(Rh*(xi-eta))*(xi^2-
```

```
eta^2), eta=-1..1), xi=1..infinity);
```

$$S := \frac{(4 Rh^2 + 6 Rh + 3) e^{-2 Rh}}{3} \quad (8.3)$$

With Rh in atomic units, we plot the magnitude of this overlap integral as a function of internuclear distance.

```
> plot(S, Rh=0..10, title="overlap integral of H2", colour=blue,
      thickness=2,
      titlefont=[TIMES,BOLD,14], labels=["R/a0","E/hartree"])
;
```



This graph with Rh/a_0 shows up to which distance $\frac{R}{2}$ wavefunction ψ_{ls} centred on nuclei A and B maintains an appreciable overlap. The result is generalizable readily to a amplitude function with a free scale parameter.

We proceed to calculate the matrix element for direct interaction for the residual potential energy. The interaction of electronic density with the nuclei belonging to the other atom respectively,

$$\iint (\psi_A(r_1) \psi_B(r_2))^2 \left(\frac{1}{r_{1B}} + \frac{1}{r_{2A}} \right) d\tau_1 d\tau_2$$

has two equal contributions that become identical; this effect becomes obvious when one relabels the integration variable. We calculate this double integral in either order.

```
> Kp := expand(2*2*Pi*Rh^3*int(int((psi(Rh*(xi+eta)))^2/(Rh*(xi-eta))
      *(xi^2-eta^2), eta=-1..1), xi=1..infinity));
```

$$Kp := -\frac{2}{(e^{Rh})^4} - \frac{1}{Rh (e^{Rh})^4} + \frac{1}{Rh} \quad (8.4)$$

```
> Kp := expand(4*Pi*Rh^2*int(int((psi(Rh*(xi+eta)))^2*(xi+eta), xi=
      1..infinity), eta=-1..1));
```

$$Kp := -\frac{2}{(e^{Rh\sim})^4} - \frac{1}{Rh\sim (e^{Rh\sim})^4} + \frac{1}{Rh\sim} \quad (8.5)$$

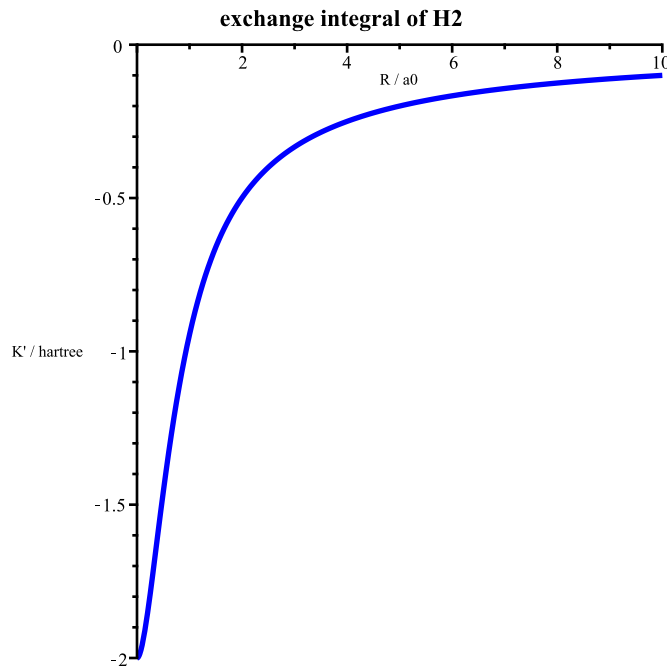
which yields the same result because the amplitude function ensures the convergence of the integral over ξ .

```
> (psi(Rh*(xi+eta)))^2;
```

$$\frac{(e^{-Rh\sim(\xi + \eta)})^2}{\pi} \quad (8.6)$$

We plot the energy, called an exchange energy or resonance energy, associated with the attraction of the electronic density from atomic centre A with nucleus B and vice versa.

```
> plot(-Kp, Rh=0..10, Kpr=-2..0, title="exchange integral of H2",
       colour=blue, thickness=2,
       titlefont=[TIMES,BOLD,14], labels=["R / a0", "K' /
       hartree"]);
```



The gain in energy due to the attraction of the electronic clouds by the opposite nuclei has a large range.

We calculate the repulsion between electrons; this calculation is complicated because the coordinates of both particles are involved. We perform the integration first with respect to coordinates of electron 2 in elliptic coordinates that use electron 1 and nucleus B (at which electron 2 is centred) as focal points;

in these coordinates separation r_{12} becomes $\frac{r_{1B}(\xi + \eta)}{2}$. We proceed with indefinite integration and substitution of boundaries.

```
> resintl := 2*Pi*r1Bh^3*int(int((psi(r1Bh*(xi-eta)))^2*(xi^2-eta^2)
    /(r1Bh*(xi+eta)), eta=-1..1), xi);
```

$$resintl := \frac{e^{-2r1Bh(\xi - 1)}}{2} + \frac{e^{-2r1Bh(\xi + 1)}}{2} \quad (8.7)$$

$$\begin{aligned}
& + \frac{e^{-2rIBh\xi + 2rIBh} (-2rIBh\xi + 2rIBh) - e^{-2rIBh\xi + 2rIBh} - 2e^{-2rIBh\xi + 2rIBh} rIBh}{4rIBh} \\
& - \frac{e^{-2rIBh\xi - 2rIBh} (-2rIBh\xi - 2rIBh) - e^{-2rIBh\xi - 2rIBh} + 2e^{-2rIBh\xi - 2rIBh} rIBh}{4rIBh} \\
& + \frac{e^{-2rIBh(\xi + 1)}}{4rIBh} - \frac{e^{-2rIBh(\xi - 1)}}{4rIBh}
\end{aligned}$$

**> resint := limit(resint1,xi=infinity)-subs(xi=1,resint1) assuming
rIBh > 0;**

$$\begin{aligned}
resint := & -\frac{e^0}{2} - \frac{e^{-4rIBh}}{2} - \frac{-e^0 - 2e^0 rIBh}{4rIBh} + \frac{-2e^{-4rIBh} rIBh - e^{-4rIBh}}{4rIBh} - \frac{e^{-4rIBh}}{4rIBh} \\
& + \frac{e^0}{4rIBh}
\end{aligned} \tag{8.8}$$

> resint := expand(resint);

$$resint := -\frac{1}{(e^{rIBh})^4} + \frac{1}{2rIBh} - \frac{1}{2rIBh(e^{rIBh})^4} \tag{8.9}$$

For the remaining integral in elliptic coordinates, the nuclei are focal points, which implies the substitution $r_{IB} = \frac{R(\xi - \eta)}{2}$.

**> 2*Pi*Rh^3*Int(Int(subs(rIBh=Rh*(xi-eta)/2,resint)*(xi^2-eta^2)*
(psi(Rh*(xi+eta)))^2,
eta=-1..1), xi=1..infinity) =
expand(2*Pi*Rh^3*int(int(subs(rIBh=Rh*(xi-eta)/2,resint)*(xi^2-
eta^2)*(psi(Rh*(xi+eta)))^2,
eta=-1..1), xi=1..infinity));**

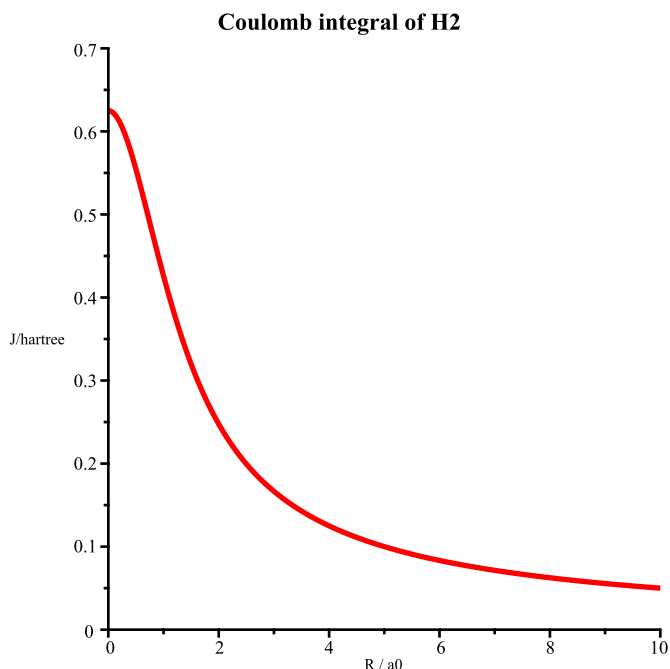
$$\begin{aligned}
2\pi Rh^3 \left(\int_1^\infty \int_{-1}^1 \frac{1}{\pi} \left(\left(-\frac{1}{\left(e^{\frac{Rh\sim(\xi-\eta)}{2}} \right)^4} + \frac{1}{Rh\sim(\xi-\eta)} \right. \right. \right. \\
\left. \left. \left. - \frac{1}{Rh\sim(\xi-\eta) \left(e^{\frac{Rh\sim(\xi-\eta)}{2}} \right)^4} \right) (-\eta^2 + \xi^2) (e^{-Rh\sim(\xi+\eta)})^2 \right) d\eta d\xi \right) = -\frac{2Rh\sim^2}{3(e^{Rh\sim})^4} \\
- \frac{3Rh\sim}{2(e^{Rh\sim})^4} - \frac{11}{8(e^{Rh\sim})^4} - \frac{1}{2Rh\sim(e^{Rh\sim})^4} + \frac{1}{2Rh\sim}
\end{aligned} \tag{8.10}$$

> J := rhs(%);

$$J := -\frac{2Rh\sim^2}{3(e^{Rh\sim})^4} - \frac{3Rh\sim}{2(e^{Rh\sim})^4} - \frac{11}{8(e^{Rh\sim})^4} - \frac{1}{2Rh\sim(e^{Rh\sim})^4} + \frac{1}{2Rh\sim} \tag{8.11}$$

We plot the resulting expansion.

```
> plot(J, Rh=0..10, Rep=0..0.7, title="Coulomb integral of H2",
      colour=red, thickness=2,
      titlefont=[TIMES,BOLD,14], labels=["R / a0",
      "J/hartree"]);
```



The latter plot shows that the coulombic repulsion between the electrons has a large range. The total direct contribution is

```
> K := simplify(-Kp + J);
```

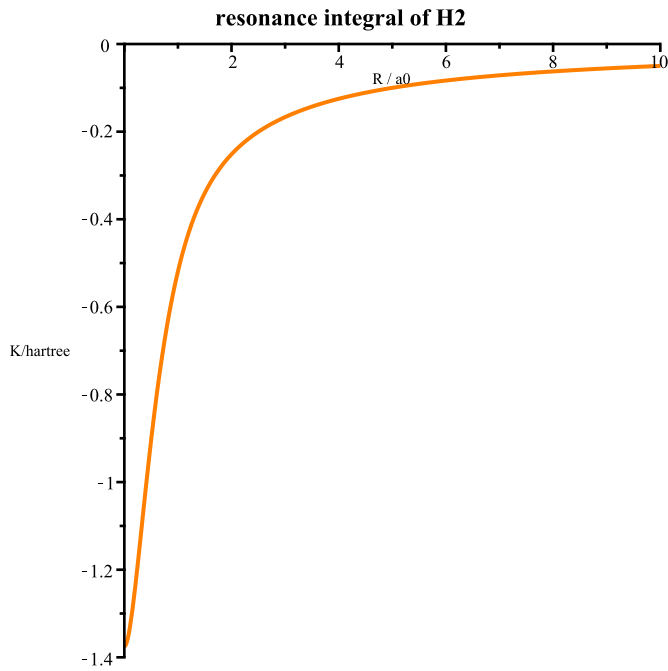
$$K := \frac{-12 + (-16 Rh^3 - 36 Rh^2 + 15 Rh + 12) e^{-4 Rh}}{24 Rh} \quad (8.12)$$

```
> expand(K);
```

$$\frac{5}{8 (e^{Rh})^4} + \frac{1}{2 Rh (e^{Rh})^4} - \frac{1}{2 Rh} - \frac{2 Rh^2}{3 (e^{Rh})^4} - \frac{3 Rh}{2 (e^{Rh})^4} \quad (8.13)$$

which we plot.

```
> plot(K, Rh=0..10, KD=-1.4..0, title="resonance integral of H2",
      colour=coral,
      titlefont=[TIMES,BOLD,14], labels=["R / a0",
      "K/hartree"]);
```



The exchange contribution is more difficult to calculate and is no longer representable in terms of solely elementary functions. The final result comprises two contributions; $A_p = A'$ is calculated in the same manner as K' .

```
> Ap := 2*exp(-4*Rh)*(1+4*Rh+16/3*Rh^2+8/3*Rh^3);
```

$$A_p := 2 e^{-4Rh} \left(1 + 4Rh + \frac{16}{3} Rh^2 + \frac{8}{3} Rh^3 \right) \quad (8.14)$$

The second contribution contains S and a similar expression S' .

```
> Sp := exp(2*Rh)*(1-2*Rh+4/3*Rh^2);
```

$$S_p := e^{2Rh} \left(1 - 2Rh + \frac{4}{3} Rh^2 \right) \quad (8.15)$$

We apply Euler's constant, represented in *Maple* as γ , and exponential integral $Ei(x)$.

```
> c_E := evalf(gamma);
```

$$c_E := 0.5772156649 \quad (8.16)$$

```
> Jx := 1/5*(exp(-4*Rh)*(25/8-23/2*Rh-12*Rh^2-8/3*Rh^3)
+3/Rh*(S^2*ln(2*Rh)+c_E*S^2+Sp^2*Ei(-8*Rh)-2*S*Sp*Ei(-4*
Rh)) );
```

$$J_x := \frac{e^{-4Rh} \left(\frac{25}{8} - \frac{23}{2} Rh - 12 Rh^2 - \frac{8}{3} Rh^3 \right)}{5} \quad (8.17)$$

$$+ \frac{1}{5 Rh} \left(3 \left(\frac{(4 Rh^2 + 6 Rh + 3)^2 (e^{-2Rh})^2 \ln(2 Rh)}{9} \right) \right)$$

$$\begin{aligned}
& + 0.06413507388 \left(4 Rh^2 + 6 Rh + 3 \right)^2 \left(e^{-2 Rh} \right)^2 + \left(e^{2 Rh} \right)^2 \left(1 - 2 Rh \right. \\
& \left. + \frac{4}{3} Rh^2 \right)^2 \text{Ei}(-8 Rh) \\
& - \frac{2 \left(4 Rh^2 + 6 Rh + 3 \right) e^{-2 Rh} e^{2 Rh} \left(1 - 2 Rh + \frac{4}{3} Rh^2 \right) \text{Ei}(-4 Rh)}{3} \Bigg)
\end{aligned}$$

> A := Jx - Ap;

$$A := \frac{e^{-4 Rh} \left(\frac{25}{8} - \frac{23}{2} Rh - 12 Rh^2 - \frac{8}{3} Rh^3 \right)}{5}$$

(8.18)

$$+ \frac{1}{5 Rh} \left(3 \left(\frac{\left(4 Rh^2 + 6 Rh + 3 \right)^2 \left(e^{-2 Rh} \right)^2 \ln(2 Rh)}{9} \right. \right.$$

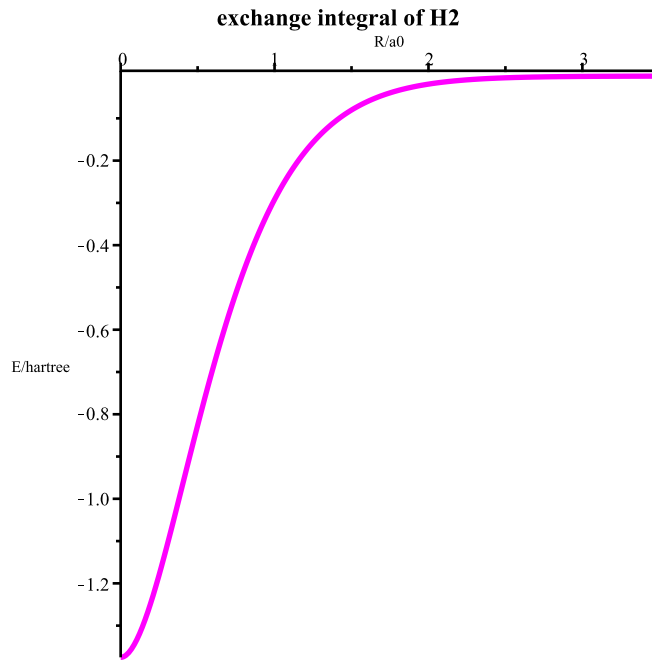
$$+ 0.06413507388 \left(4 Rh^2 + 6 Rh + 3 \right)^2 \left(e^{-2 Rh} \right)^2 + \left(e^{2 Rh} \right)^2 \left(1 - 2 Rh \right.$$

$$+ \frac{4}{3} Rh^2 \Bigg)^2 \text{Ei}(-8 Rh)$$

$$- \frac{2 \left(4 Rh^2 + 6 Rh + 3 \right) e^{-2 Rh} e^{2 Rh} \left(1 - 2 Rh + \frac{4}{3} Rh^2 \right) \text{Ei}(-4 Rh)}{3} \Bigg)$$

$$- 2 e^{-4 Rh} \left(1 + 4 Rh + \frac{16}{3} Rh^2 + \frac{8}{3} Rh^3 \right)$$

> plot(A, Rh=0..3.5, title="exchange integral of H2", colour=magenta, thickness=2, titlefont=[TIMES,BOLD,14], labels=["R/a0", "E/hartree"]);



The exchange contribution to the interaction energy has the properties that its range is not large, which is expected because as the overlap between the two atomic orbitals decreases anti-symmetrization becomes unimportant, and that in its behaviour at small distance the values appear to be large for a supposedly small effect related to the spin structure. Although one might think that spin effects are negligible as the motion is non-relativistic and the hamiltonian lacks a dependence on spin, symmetries in the amplitude functions can be powerful.

We combine all contributions separately for the singlet and triplet cases. We choose the energy of two hydrogen atoms in their electronic ground state at infinite separation as the zero of the energy scale, which corresponds to $E=0$ asymptotically for the dissociation limit, and plot together the energies of the singlet and triplet states as a function of internuclear distance.

> Etriplet := 1/(2*Rh) + (K - A)/(1 + S^2);

$$E_{\text{triplet}} := \frac{1}{2 Rh} \quad (8.19)$$

$$+ \frac{1}{\frac{(4 Rh^2 + 6 Rh + 3)^2 (e^{-2 Rh})^2}{9} + 1} \left(\frac{1}{24 Rh} (-12 + (-16 Rh^3 - 36 Rh^2 + 15 Rh + 12) e^{-4 Rh}) - \frac{e^{-4 Rh} \left(\frac{25}{8} - \frac{23}{2} Rh - 12 Rh^2 - \frac{8}{3} Rh^3 \right)}{5} \right) - \frac{1}{5 Rh} \left(3 \left(\frac{(4 Rh^2 + 6 Rh + 3)^2 (e^{-2 Rh})^2 \ln(2 Rh)}{9} \right) \right)$$

$$\begin{aligned}
& + 0.06413507388 \left(4 Rh_{\sim}^2 + 6 Rh_{\sim} + 3 \right)^2 \left(e^{-2 Rh_{\sim}} \right)^2 + \left(e^{2 Rh_{\sim}} \right)^2 \left(1 - 2 Rh_{\sim} \right. \\
& \left. + \frac{4}{3} Rh_{\sim}^2 \right)^2 \text{Ei}(-8 Rh_{\sim}) \\
& - \frac{2 \left(4 Rh_{\sim}^2 + 6 Rh_{\sim} + 3 \right) e^{-2 Rh_{\sim}} e^{2 Rh_{\sim}} \left(1 - 2 Rh_{\sim} + \frac{4}{3} Rh_{\sim}^2 \right) \text{Ei}(-4 Rh_{\sim})}{3} \left. \right) \\
& + 2 e^{-4 Rh_{\sim}} \left(1 + 4 Rh_{\sim} + \frac{16}{3} Rh_{\sim}^2 + \frac{8}{3} Rh_{\sim}^3 \right) \left. \right)
\end{aligned}$$

```

> P3 := plot(Etriplet, Rh=0..3.5, E=-0.15..0.4, colour=blue,
            thickness=2, legend="triplet state");

```

```

> Esinglet := 1/(2*Rh) + (K + A)/(1 + S^2);

```

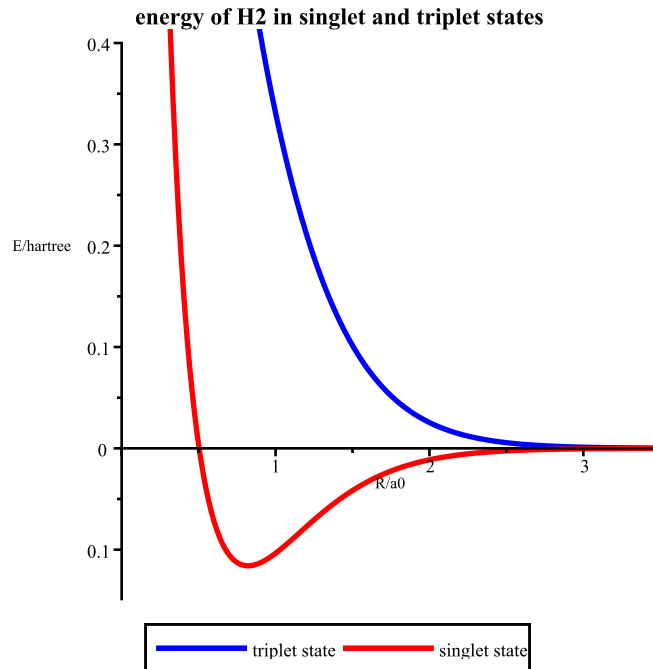
$$Esinglet := \frac{1}{2 Rh_{\sim}}$$

(8.20)

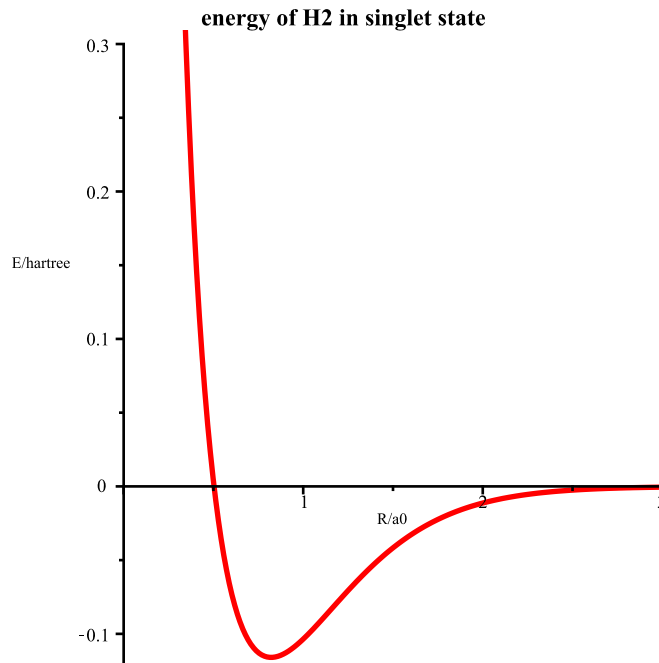
$$\begin{aligned}
& + \frac{1}{\frac{\left(4 Rh_{\sim}^2 + 6 Rh_{\sim} + 3 \right)^2 \left(e^{-2 Rh_{\sim}} \right)^2}{9} + 1} \left(\frac{1}{24 Rh_{\sim}} \left(-12 + \left(-16 Rh_{\sim}^3 - 36 Rh_{\sim}^2 \right. \right. \right. \\
& \left. \left. + 15 Rh_{\sim} + 12 \right) e^{-4 Rh_{\sim}} \right) + \frac{e^{-4 Rh_{\sim}} \left(\frac{25}{8} - \frac{23}{2} Rh_{\sim} - 12 Rh_{\sim}^2 - \frac{8}{3} Rh_{\sim}^3 \right)}{5} \\
& \left. + \frac{1}{5 Rh_{\sim}} \left(3 \left(\frac{\left(4 Rh_{\sim}^2 + 6 Rh_{\sim} + 3 \right)^2 \left(e^{-2 Rh_{\sim}} \right)^2 \ln(2 Rh_{\sim})}{9} \right. \right. \right. \\
& + 0.06413507388 \left(4 Rh_{\sim}^2 + 6 Rh_{\sim} + 3 \right)^2 \left(e^{-2 Rh_{\sim}} \right)^2 + \left(e^{2 Rh_{\sim}} \right)^2 \left(1 - 2 Rh_{\sim} \right. \\
& \left. + \frac{4}{3} Rh_{\sim}^2 \right)^2 \text{Ei}(-8 Rh_{\sim}) \\
& - \frac{2 \left(4 Rh_{\sim}^2 + 6 Rh_{\sim} + 3 \right) e^{-2 Rh_{\sim}} e^{2 Rh_{\sim}} \left(1 - 2 Rh_{\sim} + \frac{4}{3} Rh_{\sim}^2 \right) \text{Ei}(-4 Rh_{\sim})}{3} \left. \right) \left. \right)
\end{aligned}$$

$$- 2 e^{-4 R h \sim} \left(1 + 4 R h \sim + \frac{16}{3} R h \sim^2 + \frac{8}{3} R h \sim^3 \right) \Bigg)$$

```
> P1 := plot(Esinglet, Rh=0..3.5, E=-0.15..0.4, colour=red,
             thickness=2, legend="singlet state"):
> plots[display]({P1,P3}, title="energy of H2 in singlet and triplet
states",
                 titlefont=[TIMES,BOLD,14], labels=["R/a0",
"E/hartree"]);
```



```
> plot(Esinglet, Rh=0..3, E=-0.12..0.3, colour=red, title="energy of
H2 in singlet state",
       titlefont=[TIMES,BOLD,14], thickness=2, labels=
["R/a0","E/hartree"]);
```



We solve for the internuclear separation at which the minimum energy occurs for the singlet state, and evaluate the energy at that distance.

$$\begin{aligned} > \text{Rma} := \text{evalf}(\text{fsolve}(\text{diff}(\text{Esinglet}, \text{Rh}), \text{Rh}=1), 4); \\ &\quad \text{Rma} := 0.8213 \end{aligned} \quad (8.21)$$

$$\begin{aligned} > \text{Rm} := \text{evalf}(2 \cdot \text{Rma} \cdot \text{ScientificConstants}[\text{GetValue}] \\ &\quad (\text{ScientificConstants}[\text{Constant}](\text{a}[0])) \cdot \text{m}, 4); \\ &\quad \text{Rm} := 8.692 \times 10^{-11} \text{ m} \end{aligned} \quad (8.22)$$

$$\begin{aligned} > \text{evalf}(\text{eval}(\text{Esinglet}, \text{Rh}=\text{Rma}) \cdot 27.12 \cdot \text{eV}, 4); \\ &\quad -3.132 \text{ eV} \end{aligned} \quad (8.23)$$

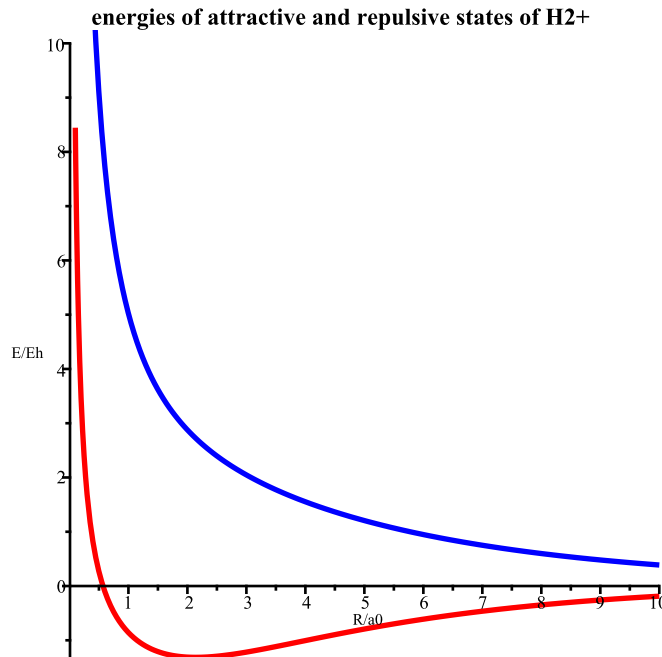
With our simple two-electron amplitude function, we estimate the binding energy to be about 3 eV and an equilibrium internuclear distance about $0.87 \cdot 10^{-10}$ m, for comparison with experimental values $D_e = 4.72$ eV and $R_e = 0.74 \cdot 10^{-10}$ m. This wavefunction is simple because it is chosen as an anti-symmetrized form of products of amplitude functions of a hydrogen atom, which corresponds to the Hartree-Fock ansatz in atoms and for which one can hope at best for a quality of total energy as achieved in the helium atom.

A simpler amplitude function for the same two states of H_2^{+} yields these energies, and their plot.

$$\begin{aligned} > \text{Energy1} := \text{R} \rightarrow -0.5 \cdot (1 + 7 \cdot \exp(-\text{R})^{0.25}) / (1 + \exp(-\text{R})) + 1/\text{R} + 0.5; \\ &\quad \text{Energy1} := \text{R} \mapsto -\frac{0.5 \cdot (1 + 7 \cdot (e^{-R})^{0.25})}{1 + e^{-R}} + \frac{1}{R} + 0.5 \end{aligned} \quad (8.24)$$

$$\begin{aligned} > \text{Energy2} := \text{R} \rightarrow -0.5 \cdot (1 - 7 \cdot \exp(-\text{R})^{0.25}) / (1 - \exp(-\text{R})) + 1/\text{R} + 0.5; \\ &\quad \text{Energy2} := \text{R} \mapsto -\frac{0.5 \cdot (1 - 7 \cdot (e^{-R})^{0.25})}{1 - e^{-R}} + \frac{1}{R} + 0.5 \end{aligned} \quad (8.25)$$


```
> plot({Energy1,Energy2}, 0.1..10, -1.4..10, colour=[red,blue],
      labels=["R/a0","E/Eh"], thickness=[2,2],titlefont=[TIMES,
      BOLD,14],
      title="energies of attractive and repulsive states of H2+
");
```



Quantum-chemical calculations imply the solution of the nuclear motion in curves or surfaces for potential energy, particularly for the ground state, from which vibration-rotational spectra might be calculated in a further step. One can investigate how molecules can store energy, particularly in the rotational motion that yields an understanding of the thermal capacity of diatomic molecular gases at temperature < 300 K. One can study also how electronic excitations occur during collisions between a particle and a molecule and how the system can undergo a transition to a state of increased energy that dissociates, such as on the repulsive curve for H_2 in its triplet state as calculated above.

3.68 energy of electrostatic interaction of two molecules

In this example, modified from material that originated with Professor Hanspeter Huber at University of Basel in Switzerland, we calculate the electrostatic energy between for two molecules, each with its charge Q , electric dipolar moment μ and quadrupolar moment Θ . All quantities have SI units.

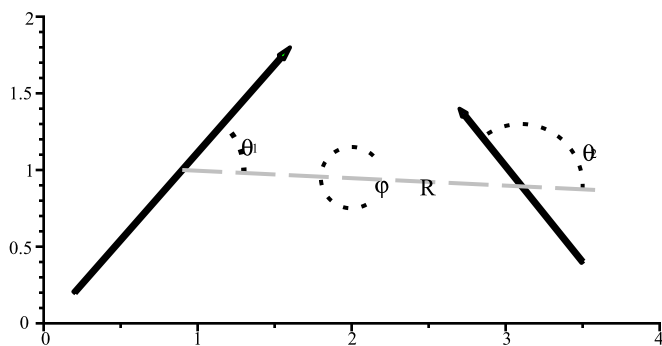
```
> restart:
> with(plots):
with(plottools):
```

The following plot depicts, schematically -- not to scale, two molecules -- molecule 1 with charge q_1 , electric dipolar moment μ_1 and electric quadrupolar moment Θ_1 at angle θ_1 , and molecule 2 with charge q_2 , electric dipolar moment μ_2 and electric quadrupolar moment Θ_2 at angle θ_2 -- represented as arrows oriented in space, distance R apart, with ϕ as the azimuthal angle relating their relative orientation about that intermolecular vector. The dashed grey line in the plot below connects the centres of the two

molecules, and angles θ_1 and θ_2 are defined with respect to that line.

```
> a1 := arc([0.9,1.0], 0.4, -0.015*Pi..0.27*Pi, thickness=2,
linestyle=dot):
a2 := arc([3.1,0.9], 0.4, -0.02*Pi..0.71*Pi, linestyle=dot):
a3 := arc([2.0,0.95], 0.2, Pi/4..7*Pi/4, linestyle=dot):
l1 := arrow([0.2, 0.2], [1.6, 1.8], 0.02, 0.045, .04, colour=
green):
l2 := arrow([3.5, 0.4], [2.7, 1.4], 0.02, 0.04, .04, colour=blue):
l3 := line([0.9,1.0], [3.6,0.869], colour=grey, linestyle=dash):
t1 := textplot([1.33,1.15, "q"], font=[SYMBOL,14]):
t2 := textplot([1.37,1.13, "1"], font=[TIMES,ROMAN,8]):
t3 := textplot([3.54,1.11, "q"], font=[SYMBOL,14]):
t4 := textplot([3.58,1.09, "2"], font=[TIMES,ROMAN,8]):
t5 := textplot([2.5,0.88, "R"], font=[TIMES,ROMAN,14]):
t6 := textplot([2.2,0.9, "j"], font=[SYMBOL,14]):
> display(a1,a2,a3,l1,l2,l3,t1,t2,t3,t4,t5,t6, scaling=constrained,
view=[0..4,0..2],
titlefont=[TIMES,BOLD,14],title="two molecules oriented in
space");
```

two molecules oriented in space



R denotes the distance/m apart.

```
> R := 300.*10^(-12);
```

$$R := 3.000000000 \times 10^{-10}$$

(9.1)

These are the two net electric charges/C .

```
> Q1 := 0.1*10^(-19);
```

(9.2)

$$Q1 := 1.000000000 \times 10^{-20} \quad (9.2)$$

```
> Q2 := 0.2*10^(-19);
```

$$Q2 := 2.000000000 \times 10^{-20} \quad (9.3)$$

These are the two electric dipolar moments/C m.

```
> mue1 := 3.0*10^(-30);
```

$$mue1 := 3.000000000 \times 10^{-30} \quad (9.4)$$

```
> mue2 := 4.0*10^(-30);
```

$$mue2 := 4.000000000 \times 10^{-30} \quad (9.5)$$

These are the two electric quadrupolar moments/C m²,

```
> Theta1 := 2.1*10^(-40);
```

$$\Theta1 := 2.100000000 \times 10^{-40} \quad (9.6)$$

```
> Theta2 := 3.1*10^(-40);
```

$$\Theta2 := 3.100000000 \times 10^{-40} \quad (9.7)$$

These are two fundamental physical constants, of Avogadro and electric permittivity ϵ_0 of free space multiplied by 4π .

```
> Avogadro := 6.02214129*10^23;
```

$$Avogadro := 6.022141290 \times 10^{23} \quad (9.8)$$

```
> FourPiEps := evalf(4*Pi*8.85418782*10^(-12));
```

$$FourPiEps := 1.112650056 \times 10^{-10} \quad (9.9)$$

These are the two angles/rad relative to the intermolecular vector as defined in the above plot.

```
> theta1 := evalf(30/180*Pi);
```

$$\theta1 := 0.5235987758 \quad (9.10)$$

```
> theta2 := evalf(45/180*Pi);
```

$$\theta2 := 0.7853981635 \quad (9.11)$$

This is the angle between the projections of the two molecules along the intermolecular vector.

```
> phi := evalf(60/180*Pi);
```

$$\phi := 1.047197551 \quad (9.12)$$

Here begin the calculations of the contributions to the total energy, which are summed at the end.

```
> LadLad := Q1*Q2/R/FourPiEps*Avogadro;      # charge, charge
```

$$LadLad := 3608.287115 \quad (9.13)$$

```
> LadDip := (Q2*mue1*cos(theta1)-Q1*mue2*cos(theta2))/R^2/FourPiEps*
Avogadro;
# charge, dipolar
moment
```

$$LadDip := 1423.905447 \quad (9.14)$$

```

> LadQuad := 1/2*(Q1*Theta2*(3*cos(theta2)^2-1) +
               Q2*Theta1*(3*cos(theta1)^2-1))/R^3/FourPiEps*Avogadro;

                                     # charge,
quadrupolar moment
               LadQuad := 681.5653440

```

(9.15)

```

> DipDip := -mue1*mue2*(2*cos(theta1)*cos(theta2) -
                    sin(theta1)*sin(theta2)*cos(phi))/R^3/FourPiEps*
Avogadro;

                                     # dipole, dipole
               DipDip := -2520.913377

```

(9.16)

```

> DipQuad := 1.5*(mue1*Theta2*(cos(theta1)*(3*cos(theta2)^2-1)
                    - 2*sin(theta1)*sin(theta2)*cos(theta2)*cos(phi))
                    - mue2*Theta1*(cos(theta2)*(3*cos(theta1)^2-1)
                    - 2*sin(theta1)*sin(theta2)*cos(theta1)*cos(phi)))
/R^4/FourPiEps*Avogadro;

                                     # dipole,
quadrupole
               DipQuad := -315.7891539

```

(9.17)

```

> QuadQuad := 0.75*Theta1*Theta2*(1-5*cos(theta1)^2-5*cos(theta2)^2
                    +17*cos(theta1)^2*cos(theta2)^2+2*sin(theta1)^2*sin(theta2)
                    ^2*cos(phi)^2
                    -16*sin(theta1)*sin(theta2)*cos(theta1)*cos(theta2)*cos(phi))
/R^5/FourPiEps*Avogadro;

                                     # quadrupole,
quadrupole
               QuadQuad := -59.21977232

```

(9.18)

```

> TotalEnergy := LadLad + LadDip + LadQuad + DipDip + DipQuad +
QuadQuad;

               TotalEnergy := 2817.835603

```

(9.19)

The total energy is hence $2817.8 \text{ J mol}^{-1}$.

e3.62 exercise

For fixed distance R , adjust angles θ_1 , θ_2 and ϕ to maximize the total energy according to the electrostatic energy between two molecules, each with its charge Q , electric dipolar moment μ and quadrupolar moment Θ .

3.7 numerical solutions for energies of vibrational states

3.71 numerical solution of Schroedinger's equation for potential energy I

This set of procedures originated with Markus Mueller and Professor Hanspeter Huber in the Institut für Physikalische Chemie, Universität Basel, 2000 July, as modified and extended for *Maple* by J. F. Ogilvie, appears here with their kind permission.

> **restart:**

The renormalised Numerov method is widely used to solve the one-dimensional eigenvalue problem for a given function for potential energy. This function can either be a model, such as a line segment or harmonic, or result from a fit to quantum-chemically calculated electronic energies. The crucial idea is to fix the eigenvalue and to solve the differential equation. If the resulting solution be not an eigenfunction, the deviation serves as a guide to alter the eigenvalue.

Here is Schrodinger's equation in atomic units, in which Dirac's constant and the electronic rest mass m_e equal unity.

$$-\frac{\left(\frac{d^2}{dx^2} \psi(x)\right) \cdot 1}{2\mu} + V(x) \psi(x) = E \psi(x)$$

With this transformation, $u''(x) = f(x) u(x)$ with $u(x) = u = \psi(x)$ and $f(x) = 2\mu (V(x) - E)$ and forming a Taylor series about x ,

$$u(x+h) = u + h u' + \frac{1}{2} h^2 u'' + \frac{1}{6} h^3 u''' + \frac{1}{24} h^4 u'''' + \frac{1}{120} h^5 u''''' + \frac{1}{720} h^6 u'''''' + \dots$$

we obtain

$$I) \quad [u(x+h) + u(x-h)] = 2u + h^2 u'' + \frac{1}{12} h^4 u'''' + \frac{1}{360} h^6 u'''''' + \dots$$

taking the second derivative of I) and multiplying with $-\frac{1}{12} h^2$,

$$II) \quad -\frac{1}{12} h^2 [u''(x+h) + u''(x-h)] = -\frac{1}{6} h^2 u'' - \frac{1}{12} h^4 u'''' - \frac{1}{144} h^6 u'''''' + \dots$$

and combining the latter two results yields

$$I) + II) \quad [u(x+h) + u(x-h)] - \frac{1}{12} h^2 [u''(x+h) + u''(x-h)] = 2u + \frac{5}{6} h^2 u'' - \frac{1}{240} h^6 u'''''' + \dots$$

Replacing u'' by the differential equation $u'' = fu$, as above,

$$[u(x+h) + u(x-h)] - \frac{1}{12} h^2 [f(x+h) u(x+h) + f(x-h) u(x-h)] = 2u + \frac{5}{6} h^2 fu - \dots$$

contracting to

$$\left[1 - \frac{1}{12} h^2 f(x+h)\right] u(x+h) + \left[1 - \frac{1}{12} h^2 f(x-h)\right] u(x-h) = (2 + 10 \frac{1}{12} h^2 f) u - \dots$$

neglecting terms of order higher in h , and using this abbreviation,

$$T(x) = \frac{1}{12} h^2 f(x)$$

we obtain this recurrence formula

$$[1 - T(x+h)] u(x+h) + [1 - T(x-h)] u(x-h) = [2 + 10 T(x)] u(x)$$

With the following abbreviations,

$$F(x) = [1 - T(x)] u(x) \quad \text{and} \quad U(x) = \frac{2 + 10 T(x)}{1 - T(x)}$$

$$F(x + h) = U(x) F(x) - F(x - h)$$

$$R(x) = \frac{F(x + h)}{F(x)}$$

we obtain this two-term recurrence formula,

$$R(x) = U(x) - R^{-1}(x - h)$$

Similarly with

$$R^0(x) = \frac{F(x - h)}{F(x)}$$

we obtain an analogous decreasing recurrence relation

$$R^0(x) = U(x) - 1 / R^0(x + h)$$

We begin integration between x_0 and x_1 with the latter recurrence formula at x_1 , for which $R^* = \infty$, and continue until R is about 1, i.e. the derivative of ψ is about 0. This point is selected as the meeting point x_M . We repeat this procedure with the first recurrence formula from x_0 , for which $R = \infty$, until x_M , and we count the number of times that $R < 0$, which yields the number of nodes. If the two functions meet, i.e. if

$$D E = R(x_M + h) - R(x_M) = 0,$$

energy E is an eigenvalue, and function Ψ calculated from R or R^* is an eigenfunction; otherwise the number of nodes and $D E$ produce criteria how E must be altered.

These articles provide further information about the mathematical background of the renormalized Numerov method:

J. M. Blatt, J. Comput. Phys. 1, 382-396 (1967)

B. R. Johnson, J. Chem. Phys. 67, 4086-4093 (1977)

> Digits := 24:

This parameter **max_n_dim** of the calculation should normally not be altered; see also **improvement of the accuracy of the calculation** below.

> max_n_dim := 2000:

speedup := 10:

The latter variable should normally not be altered; with its use each eigenvalue is first calculated with an array size, which is *speedup* times smaller than *max_n_dim*, so to accelerate the calculation. The eigenvalue thus found becomes then used as initial value for the subsequent calculation with the full array size.

1) Select a potential

Program section **Selectable potentials** presents selected functions for potential energy.

2) Copy the name of a selected procedure for potential energy.

Open the subsection for the potential-energy function that you prefer for the calculation. Copy the procedure name (e.g. **harmonic_potential :=**).

3) Insert the potential procedure name.

Replace the name in the next line to the right of the assignment operator by the selected name.

mue --> default reduced mass

> **mue := 1:**

Default value for reduced mass in atomic unit, i.e. m_e ; individual values are set in the procedures for functions of potential energy for physically meaningful potentials like Ne_2 , H_2 etc.

selectable potentials

There exist below 20 selectable functions of potential energy that operate satisfactorily; one of these must be activated to proceed beyond this point. In My Potential there is a template that is applicable to generate a procedure for a further function. After activating a particular function, state the exact name of this procedure in the first command after this list.

box (1)

A particle is confined to a line segment of length 2000 units with infinitely great potential energy at the bounds and zero potential energy between; $\mu = 1$ by default.

```
> box_potential := proc(max_n,Acquisition,focus::evaln,mue::evaln,
    xscale::evaln,potentialscaling::evaln,func_modulator::evaln,
    unam::evaln)
    local k;
    focus := 1:
    xscale := 1:
    func_modulator := 5:
    potentialscaling := 1:
    unam := "microhartree";
    for k from 1 to max_n do
        Acquisition[Potential,k] := 0;
    end do;
end proc;
```

topf (2)

A particle is confined to a line segment of length 1200 units between 400 and 1600 units with zero potential energy, or to a greater length 2000 units between 0 and 2000 with a finite potential energy equal to 200 units at the bounds between 0 and 400 and between 1600 and 2000 and zero potential energy between 400 and 1600; $\mu = 1$ by default.

```
> topf_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,x,side,stature;
    side := 0.2;
    stature := 200;
    focus := round(max_n_dim/max_n_dim);
    potentialscaling := 1;
    unam := "microhartree";
    func_modulator := 10;
    xscale := 1;
```

```

for k from 1 to max_n_dim do
  x := (k / max_n_dim)*eval(xscale);
  if (x < side) or (x > 1.0 - side) then
    Acquisition[Potential,k] := stature;
  else
    Acquisition[Potential,k] := 0;
  end if;
end do;
end proc:

```

wall (3)

A particle is confined to a line segment of length 2000 with zero potential energy, except that a wall or barrier exists of stature 100 units between 980 and 1020 units; $\mu = 1$ by default.

```

> wall_potential := proc(max_n_dim,Acquisition,focus::evaln,
  mue::evaln,xscale::evaln,potentialscaling::evaln,
  func_modulator::evaln, unam::evaln)
  local k,x,width_of_half_wall,stature,wo,xshift;
  width_of_half_wall := 0.02;
  stature             := 100;
  wo                  := 0.5;
  xshift              := 0.0;
  xscale              := 1;
  focus               := round(max_n_dim/max_n_dim);
  func_modulator      := 10;
  potentialscaling    := 1;
  unam := "microhartree";
  for k from 1 to max_n_dim do
    x := (k / max_n_dim - xshift) * eval(xscale);
    if (x > wo - width_of_half_wall)
      and (x < wo + width_of_half_wall) then
      Acquisition[Potential,k] := stature;
    else
      Acquisition[Potential,k] := 0;
    end if;
  end do;
end proc:

```

harmonic 1 (4)

A particle is subject to a potential that is proportional to its local displacement from an equilibrium position, set at 1000 units; $\mu = 1$ by default. This function of potential energy has therefore a parabolic form.

```

> harmonic1_potential := proc(max_n_dim,Acquisition,focus::evaln,

```



```

        mue::evaln,xscale::evaln,potentialscaling::evaln,
        func_modulator::evaln, unam::evaln)
local force_constant,xshift,const,x,k;
force_constant := 5839.077;
mue := 1/0.00054858;
xshift := 0.5;
xscale := 1;
focus := 1;
potentialscaling := 2000;
unam := "microhartree";
func_modulator := 40;
const := 0.5 * force_constant;
for k to max_n_dim do
    x := (k / max_n_dim - xshift)*eval(xscale);
    Acquisition[Potential,k] := const * x^2;
end do;
end proc:

```

harmonic 2 (5)

```

> harmonic2_potential := proc(max_n_dim,Acquisition,focus::evaln,
        mue::evaln,xscale::evaln,potentialscaling::evaln,
        func_modulator::evaln, unam::evaln)
local force_constant,xshift,const,x,k; # &&&
force_constant := 5839.08;
mue := 1/0.00054858;
xshift := -0.6;
xscale := 1;
focus := 1;
potentialscaling := 5000;
unam := "microhartree";
func_modulator := 100;
const := 0.5 * force_constant;
for k to max_n_dim do
    x := (k / max_n_dim - xshift)*eval(xscale);
    Acquisition[Potential,k] := const * (x^2 + 1/x^2)/2 - 2920.;
end do;
end proc:

```

invexp (6)

This function for potential energy is symmetric about a point at 1000 units, at which the depth of the potential energy is -1000 units relative to zero at the asymptotic limits of the exponential function at either side, according to an inverse exponential function of form $e^{-|\beta x|}$. The number of bound states

supported is finite.

```
> invexp_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,x,a,b,xshift;
    a := 1000;
    b := 5;
    mue := 1/0.00054858;
    xshift := 0.5;
    xscale := 1;
    focus := round(max_n_dim/max_n_dim);
    potentialscaling := 2000;
    unam := "microhartree";
    func_modulator := 50;
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * eval(xscale);
        Acquisition[Potential,k] := -a * exp(-abs(b * x));
    end do;
end proc;
```

Morse (7)

Morse's function for potential energy, which was the first to be solved exactly according to quantum mechanics that supports vibrational states of finite number, employs exponential functions for both repulsive and attractive interactions, according to $V_M = D \left(e^{-2\beta x} - 2 e^{-\beta x} \right)$. For particular parameters applied here, the minimum potential energy occurs at $\sim 0.2 * 2000 = 400$ units, and the depth there below the dissociation limit for $\beta = 1.25194$ is 57160 units; these parameters are roughly suitable for HF, with the energy scale on the plot in cm^{-1} , not $\mu\text{hartree}$.

```
> morse_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,x,D,b,tmpexp,xshift;
    D := 57160;
    b := 1.25194;
    mue := 1/0.00054858;
    xshift := 0.2;
    xscale := 4;
    focus := round(max_n_dim/18);
    potentialscaling := 2*109737.31568549;
    unam := "1/cm";
    func_modulator := 3000;
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * eval(xscale);
```

```

        Acquisition[Potential,k] := D * (exp(-2*b*x) - 2*exp(-b*x));
    end do;
end proc:

```

```

> morseo_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,x,D,b,tmpexp,xshift;    # old Morse potential
    D := 5.716;
    b := 0.15194;
    xshift := 0.2;
    xscale := 40;
    focus := round(max_n_dim/18);
    potentialscaling := 1;
    unam := "microhartree";
    func_modulator := 1;
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * eval(xscale);
        tmpexp := exp(-b * x);
        Acquisition[Potential,k] := D * ((tmpexp^2) - 2 * tmpexp);
    end do;
end proc:

```

double 1 (8)

A particle is subject to a function of potential energy, symmetric about a position at 1000 units, that possesses two equal minima, with two states below the barrier or hump of stature 100 units between the two minima. As the form of this function is parabolic apart from that central and symmetric hump, the number of bound states is infinite.

```

> double1_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,b,c,sqr_relative_length,xshift,x;
    b := 15000;
    c := 2400;
    xshift := 0.503;
    xscale := 1;
    focus := 1;
    func_modulator := 20;
    potentialscaling := 1;
    unam := "microhartree";
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * eval(xscale);
        sqr_relative_length := x^2;
    end do;
end proc:

```

```

        Acquisition[Potential,k] := b * sqr_relative_length^2 - c *
            sqr_relative_length + 100;
    end do;
end proc:

```

double 2 (9)

A particle is subject to a function of potential energy, symmetric about a position at 1000 units, that possesses two equal minima, with two states below the barrier or hump of stature 140 units between the two minima at -400 units. As the form of this function is gaussian, $-e^{\beta x^2}$, having two such functions overlapping sufficiently to provide the double minimum, arising from that central and symmetric hump, the number of bound states is finite, and for states of positive energy the particle is free.

```

> double2_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,x,De,w,xo,xshift;
    De := 400;
    w := 50;
    xo := 0.15;
    xshift := 0.5;
    xscale := 1;
    focus := round(max_n_dim/max_n_dim);
    func_modulator := 20;
    potentialscaling := 1;
    unam := "microhartree";
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * eval(xscale);
        Acquisition[Potential,k] := -De * (exp(-w * (x - xo)^2)
            + exp(-w * (x + xo)^2));
    end do;
end proc:

```

NH3 inversion (10)

This function of potential energy that possesses a double minimum with a central hump of stature 10 unit at 1000 units along the abscissal axis. The function is polynomial in form but with only even powers.

```

> NH3inv_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,x,xshift,a0,a2,a4,a6,a8;
    a0 := -56.3887;
    a2 := -0.1478;
    a4 := 0.4248;

```

```

a6 := -0.2203;
a8 := 3.4985e(-2);
# calculating reduced mass using N -> 14/m_e = 25532.71 au as m1
#                               and (3*H) -> 3*1/m_e = 5512.08 au as m2
#                               with mue = m1*m2/((m1 + m2)*m_e)
mue := 4533.4;
xscale := 1.2;
xshift := 0.503;
focus := round(max_n_dim/max_n_dim);
func_modulator := 3;
potentialscaling := 1000;
unam := "microhartree";
for k from 1 to max_n_dim do
  x := (k / max_n_dim - xshift) * xscale;
  Acquisition[Potential,k] := potentialscaling * (a0
    + (a2 * x^2) + (a4 * x^4)
    + (a6 * x^6) + (a8 * x^8));
end do;
end proc:

```

periodic (11)

A particle is subject to a function of potential energy having six minima, unequal but symmetric about the centre, such as for a 'conduction' electron in a metal. Each minimum is formed from an inverse exponential function of the type in potential-energy function 5 above.

```

> periodic_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
  local k,p,x,D,w,xo,xshift,half_well_numb,Doo,xoo,woo;
  D := 400;
  w := 50;
  xo := 0.15;
  half_well_numb := 3;
  xshift := 0.5;
  xscale := 1;
  focus := round(max_n_dim/max_n_dim);
  func_modulator := 20;
  potentialscaling:= 1;
  unam := "microhartree";
  Doo := 1.4 * D;
  xoo := 2 * xo / sqrt(half_well_numb);
  woo := 0.08 * w * half_well_numb;
  for k from 1 to max_n_dim do

```

```

x := (k / max_n_dim - xshift) * eval(xscale);
Acquisition[Potential,k] := 0;
for p from 1 to half_well_numb do
  Acquisition[Potential,k] := evalf(Acquisition[Potential,k]
    - Doo * (exp(-woo * abs(x - (p - 0.5) * xoo))
    + exp(-woo * abs(x + (p - 0.5) * xoo))));
end do;
end do:
end proc:

```

H2plus (12)

This function for potential energy of H_2^+ is formed to resemble the exact result of a quantum-mechanical calculation. The form of this function is $V_{\text{H}_2^+} = D_e \left(\left(1 + \gamma \beta^3 R^3 \right) e^{-2\beta R} - 2 e^{-\beta R} \right)$, with parameters D_e, β, γ chosen appropriately. The reduced mass in this case is $\frac{M_H}{2 m_e}$.

```

> H2Plus_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
  local k,x,D,b,c,tmpexp,xshift;
  D := 0.109105;
  b := 0.707738;
  c := -0.379001;
  mue:= 919.783;
  xshift := 0.10819155;
  xscale := 20;
  focus := round(max_n_dim/20);
  func_modulator := 1000;
  potentialscaling := 2*109737.31568549; # for cm^-1 #1.0e6;
  unam := "1/cm";
  for k from 1 to max_n_dim do
    x := (k / max_n_dim - xshift) * eval(xscale);
    tmpexp := exp(-b * x);
    Acquisition[Potential,k] := potentialscaling *(D * ( -
      2 * tmpexp + (1 + c * b^3 * x^3) * tmpexp^2));
  end do;
end proc:

```

HDplus (13)

This function for potential energy of HD^+ is formed to resemble the exact result of a quantum-mechanical calculation. The form of this function is $V_{\text{HD}^+} = D_e \left(\left(1 + \gamma \beta^3 R^3 \right) e^{-2\beta R} - 2 e^{-\beta R} \right)$, with parameters D_e, β, γ chosen appropriately but the same as for H_2^+ . The reduced mass in this case is

$$\frac{M_H M_D}{(M_H + M_D) m_e}.$$

```
> HDPlus_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,x,D,b,c,tmpexp,xshift;
    D      := 0.109105;
    b      := 0.707738;
    c      := -0.379001;
    mue    := 1226.37733;
    xshift := 0.10819155;
    xscale := 20;
    focus  := round(max_n_dim/20);
    func_modulator := 1000;
    potentialscaling := 2*109737.31568549; # for cm^-1 # 1.0e6;
    unam := "1/cm";
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * eval(xscale);
        tmpexp := exp(-b * x);
        Acquisition[Potential,k] := potentialscaling *(D * ((tmpexp^2)
-
        2 * tmpexp + c * b * x * ((b * x)^2) * (tmpexp^2))
    ;
    end do;
end proc;
```

D2plus (14)

This function for potential energy of D_2^+ is formed to resemble the exact result of a quantum-mechanical calculation. The form of this function is $V_{\text{D}_2^+} = D_e \left(\left(1 + \gamma \beta^3 R^3 \right) e^{-2\beta R} - 2 e^{-\beta R} \right)$, with parameters D_e, β, γ chosen appropriately but the same as for H_2^+ . The reduced mass in this case is

$$\frac{M_D}{2 m_e}.$$

```
> D2Plus_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
```

```

local k,x,D,b,c,tmpexp,xshift;
D      :=      0.109105;
b      :=      0.707738;
c      :=      -0.379001;
mue    := 1839.566;
xshift :=      0.10819155;
xscale :=      15;
focus := round(max_n_dim/20);
func_modulator :=      1000;
potentialscaling := 2*109737.31568549; # for cm^-1 # 1.0e6;
unam := "1/cm";
for k from 1 to max_n_dim do
  x := (k / max_n_dim - xshift) * eval(xscale);
  tmpexp := exp(-b * x);
  Acquisition[Potential,k] := potentialscaling *(D * ((tmpexp^2)
-
      2 * tmpexp + c * b * x * ((b * x)^2) * (tmpexp)^2));
end do;
end proc:

```

neon1 (15)

This function for potential energy of interaction of two neon atoms supports two vibrational states, i. e. $v=0$ and $v=1$, below the dissociation limit; further states for which energies are calculated are translational states with free motion of the reduced mass in a space for which the internuclear separation exceeds the repulsive wall. This function employs as a repulsive term a single exponential function proportional to e^{-R} , whereas to represent the attraction there are terms in inverse distance -- $\frac{1}{R^6}$, $\frac{1}{R^8}$, $\frac{1}{R^{10}}$, $\frac{1}{R^{12}}$, with multiplicands of the former three terms having negative values, representing interactions between dipoles, quadrupoles, et cetera.

```

> neon1_potential := proc(max_n_dim,Acquisition,focus::evaln,
      mue::evaln,xscale::evaln,potentialscaling::evaln,
      func_modulator::evaln, unam::evaln)
  local k,x,xshift,a1,a2,a3,a4,a5,a6;
  a1 := 148.735474;
  a2 := 2.3520917;
  a3 := 3.749754e4;
  a4 := 5.160348e3;
  a5 := 8.3724779;
  a6 := 6.8028878;
  mue := 20^2/40/5.4858e-4; # 18395.667;
  xscale := 10.0;
  xshift := -0.4;

```



```

potentialscaling := 2*109737.31568549; # for cm^-1 #1.0e6;
unam := "1/cm";
func_modulator := 20;
focus := round(max_n_dim/8);
for k from 1 to max_n_dim do
    x := (k / max_n_dim - xshift) * eval(xscale);
    Acquisition[Potential,k] := potentialscaling *
        (a1*exp(-a2 * x) + a3/x^12 - a4/x^10 - a5/x^8 - a6/x^6);
end do;
end proc:

```

neon2 (16)

This function for potential energy of interaction of two neon atoms supports two vibrational states, i. e. $v=0$ and $v=1$, below the dissociation limit; further states for which energies are calculated are translational states with free motion of the reduced mass in a space for which the internuclear separation exceeds the repulsive wall. This function employs as repulsive term three exponential functions proportional of form $\alpha e^{-\beta R^2}$, whereas to represent the attraction there are terms in inverse distance -- $\frac{1}{R^6}$, $\frac{1}{R^8}$ with appropriate multiplicands having negative values but the multiplicand of $\frac{1}{R^{10}}$ having a positive value, representing interactions between dipoles, quadrupoles, et cetera.

```

> neon2_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,x,xshift,a11,a12,a13,a14,a15,a16,a17,a18,a19;
    a11      :=      1.174594e6;
    a12      :=      0.355905;
    a13      :=      1.325805e5;
    a14      :=      0.214065;
    a15      :=     976.69318;
    a16      :=      0.105389;
    a17      :=      5.412694e9;
    a18      :=     -3.512808e8;
    a19      :=     -3.964331e6;
    mue      :=     20^2/40/5.4857990945e-4; # 18395.667;
    xscale   :=      10.0;
    xshift   :=     -0.4;
    focus    := round(max_n_dim/8);
    func_modulator :=    10;
    potentialscaling :=  2*109737.31568549; # for cm^-1 #1.0e6;
    unam     := "1/cm";
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * xscale;

```

```

        Acquisition[Potential,k] := potentialscaling/10^6* (a11 *
            exp(-a12*x^2) + a13*exp(-a14*x^2) + a15*exp(-a16*x^2)
            + a17/x^10 + a18/x^8 + a19/x^6);
    end do;
end proc:

```

neon3 (17)

This function for potential energy of interaction of two neon atoms, from source HFDB, supports two vibrational states, i.e. $v=0$ and $v=1$, below the dissociation limit; further states for which energies are calculated are translational states with free motion of the reduced mass in a space for which the internuclear separation exceeds the repulsive wall. This function employs as a repulsive term a single exponential function proportional to e^{-R} , whereas to represent the attraction there are terms in inverse distance -- $\frac{1}{R^6}$, $\frac{1}{R^8}$, $\frac{1}{R^{10}}$, with their three multiplicands having negative values, representing interactions between dipoles, quadrupoles, et cetera. There is also a switching term that causes factors of these dispersion terms to depend whether R is greater or less than a particular value.

```

> neon3_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
local k,x,xshift,a21,a22,a23,a24,
    a25,a26,a27,a28,a29,Fx;

    a21      := 133.8;
    a22      := 8.9571795e5;
    a23      := 13.86434671;
    a24      := 0.12993822;
    a25      := 0.24570703;
    a26      := 0.53222749;
    a27      := 1.21317545;
    a28      := 1.36;
    a29      := 5.841;
    mue      := 20^2/40/5.4858e-4; # 18395.667;
    xscale   := 10.0;
    xshift   := -0.4;
    focus    := round(max_n_dim/10);
    func_modulator := 20;
    potentialscaling := 2*109737.31568549; # for cm^-1 #1.0e6;
    unam     := "1/cm";
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * eval(xscale) / a29;
        if x < a28 then
            Fx := exp(-(a28 / x - 1)^2);
        else

```

```

        Fx := 1;
    end if;
    Acquisition[Potential,k] := potentialscaling/1.0e6*
        (a21 * (a22 * exp(-a23 * x - a24 * x^2) -
        Fx * (a25/x^10 + a26/x^8 + a27/x^6)));
    end do;
end proc:

```

neon4 (18)

This function for potential energy of interaction between two neon atoms has an attractive term proportional to $\frac{1}{R^6}$ and a repulsive term proportional to e^{-R} .

```

> neon4_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)
    local k,Invsqr,Invhoch4,Invhoch6,x,x2,xshift,a16,alpha,
        epsilon,r_asterix;
    alpha      := 14.66;
    epsilon     := 115.419;
    r_asterix   := 5.897836;
    mue         := 18395.667;
    xscale      := 10.0;
    xshift      := -0.4;
    focus       := round(max_n_dim/9);
    func_modulator := 20;
    potentialscaling := 2*109737.31568549; # for cm^-1 #1.0e6;
    unam := "1/cm";
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * eval(xscale) / r_asterix;
        Acquisition[Potential,k] := potentialscaling/1.0e6*epsilon
            * (6*exp(alpha*(1 - x)) - alpha/x^6)/(alpha - 6);
    end do;
end proc:

```

argon1 (19)

This function of potential energy between two argon atoms, which results from quantum-chemical calculations of molecular electronic structure by Woon reported in Journal of Chemical Physics, 100, 2838 -- 2850 (1994), has a form resembling Morse's function, comprising two exponential terms of form $e^{-\beta x}$ with separate values of β and numerical multiplicands of opposite signs.

```

> argon1_potential := proc(max_n_dim,Acquisition,focus::evaln,
    mue::evaln,xscale::evaln,potentialscaling::evaln,
    func_modulator::evaln, unam::evaln)

```

```

local k,x,xshift,wo1,wo2,wo3,wo4,wo5,wo6,ewo54,ewo64;
wo1      :=      0.4228;
wo2      :=      0.6060;
wo3      :=      0.6627;
wo4      :=      7.1714;
mue      := 36423.46;
xscale   :=      10;
xshift   :=     -0.4;
focus   :=      round(max_n_dim/4.5);
func_modulator :=      20;
potentialscaling := 2*109737.31568549; # for cm^-1  #1.0e6;
unam := "1/cm";
wo5      := sqrt(wo2*wo3/((wo2 + 1.0)*wo1));
wo6      := sqrt((wo2 + 1.0)*wo3/wo2/wo1);
ewo64 :=      wo1*wo2*exp(wo6*wo4);
ewo54 :=      wo1*(wo2 + 1.0)*exp(wo5*wo4);
for k from 1 to max_n_dim do
  x := (k / max_n_dim - xshift) * eval(xscale);
  Acquisition[Potential,k] := potentialscaling/1.0e3*(
    ewo64*exp(-wo6*x) - ewo54*exp(-wo5*x));
end do;
end proc:

```

argon2 (20)

This function of potential energy between two argon atoms, which results from inversion of data for the second virial coefficient of gaseous argon between 100 and 1000 K, has a form based on Kihara's function,

$$V(R) := 4 \epsilon \left(\left(\frac{\sigma - 2a}{R - 2a} \right)^{12} - \left(\frac{\sigma - 2a}{R - 2a} \right)^6 \right)$$

comprising two terms of form inverse power of R with opposite signs. The three parameters are ϵ the depth of the minimum, relative to zero for separate atoms, σ the internuclear distance at which $V(R)$ alters from negative to positive sign, approximately $00.9 R_e$ with R_e the separation at the minimum energy, and a the radius of an impenetrable core of each argon atom. This function is an extension of the function of Lennard-Jones,

$$V(R) := 4 \epsilon \left(\left(\frac{\sigma}{R} \right)^{12} - \left(\frac{\sigma}{R} \right)^6 \right)$$

that has only two parameters but of similar physical interpretation.

```

> argon2_potential := proc(max_n_dim,Acquisition,focus::evaln,
  mue::evaln,xscale::evaln,potentialscaling::evaln,
  func_modulator::evaln, unam::evaln)
local k,x,xshift,sigma,epsilon,a,s2a;
epsilon      := 109.42;

```

```

sigma      := 3.2707;
a          := 0.47661;
s2a        := 3.2707-0.47661;
mue        := 36423.46;
xscale     := 14;
xshift     := -0.5;
focus     := round(max_n_dim/40.);
func_modulator := 10;
potentialscaling := 2*109737.31568549; # for cm-1 #1.0e6;
unam := "1/cm";
for k from 1 to max_n_dim do
  x := (k / max_n_dim - xshift) * eval(xscale);
  Acquisition[Potential,k] := potentialscaling/4000*
    4*epsilon*((s2a/(x-a))12-(s2a/(x-a))6);
end do;
end proc;

```

My Potential (21)

This example potential is the harmonic one and should help the user to implement his own potential by altering the potential function and then adjusting the different parameters step by step until a satisfying output is generated.

```

> my_potential := proc(max_n_dim,Acquisition,focus::evaln,
mue::evaln,
  xscale::evaln,potentialscaling::evaln,
func_modulator::evaln,
  unam::evaln)
  local k,x,xshift,force_constant; # local parameters
# Add here additional parameters of your potential, and add their
# names to the list of local parameters.
  force_constant := 4000;
# reduced mass
  mue := 1;
# display parameters
  xscale := 1;
  xshift := 0.5;
  focus := round(max_n_dim/max_n_dim);
# parameter to increase (> 1) or to decrease (< 1, > 0)
# the amplitude of the wavefunction
  func_modulator := 20;
# parameter to alter the energy of the potential by multiplication
# with a constant -> numerical adjustment to empirical
  levels

```

```

    potentialscaling := 1;
    unam := "microhartree";
# storage of the potential function into an array (continuous ->
# discrete)
    for k from 1 to max_n_dim do
        x := (k / max_n_dim - xshift) * eval(xscale);
        Acquisition[Potential,k] := 0.5*force_constant*x^2;
    end do;
end proc;
> selected_potential := morse_potential;
    selected_potential := morse_potential

```

(11.1)

4) Select the 'orbitals' (eigenfunctions) that you seek to calculate.

In the following lines you find assignments of two variables **from_orbital := 1;** and **to_orbital :=**

5: -- enter your desired range.

```

> from_orbital := 1;
    to_orbital    := 5;
                                from_orbital := 1
                                to_orbital := 5

```

(11.2)

5) Execute the program

In menu "Edit", select "Execute Selection" -> "".

improvement of the calculation accuracy

Parameter **max_n_dim** ("globals and arrays") determines the array width. Increasing **max_n_dim** (default 2000) yields more accurate energies and higher resolution in the output display. Note that the greater is **max_n_dim**, the greater is the duration of calculation -- find a compromise! Our default values are set for teaching, not research! Be aware that an accurate calculation might also require a larger *xscale* (see below), especially for higher orbitals, because the potentials go to infinity at the end of the array.

Amplitude of the wavefunction

For more informative displays it is sometimes useful to vary the wavefunction amplitude. In every procedure for potential energy you can find the parameter **func_modulator**. Increasing the value of this parameter increases the amplitude of a wavefunction.

Calculation of energies of states of other function for potential energy

In the section **Selectable potentials** a template potential named "My Potential" is found at the end. A regular and continuous function can be readily implemented with this potential. The parameters at the beginning must be initialized. The function with the variable **x** is inserted at the line **Acquisition [Potential,k] := .** For non-continuous potentials use the potential **Topf (7)** as a template. At this point you are already able to initiate a calculation, but you must accept that the output might not be informative because of some scale and display parameters that must be altered.

local parameters

Do not adjust the parameters *k*, *x* and *xshift*, just add your own parameters necessary to describe the potential.

reduced mass

Calculation of realistic problems in molecule spectrometry require finite masses for additional parameters. Generally for vibrational spectroscopy one uses the *reduced mass* to describe the system quantitatively:

$$\mu = \frac{m_1 m_2}{m_1 + m_2}$$

For mass unit we employ electronic mass m_e , consistent with the energy output of this program being in **microhartree**; for physically meaningful cases such as Ne dimer, H_2 etc. the reduced mass in unified atomic mass unit must be divided by the electronic mass in that unit, i.e. $5.4857990945 \cdot 10^{-4}$. The name or symbol to designate reduced mass in the procedures is *mue*.

display parameters (global)

xscale: *default value = 1*

With the definition

$$x = \left(\frac{i}{\text{max_n_dim}} - \text{xshift} \right) \text{xscale}$$

the potential is plotted from $x = 0$ to $x = 1$ with the values $\text{xshift}=0$ and $\text{xscale}=1$. For realistic problems one has generally to scale the x -axis. For example the **Neon2** potential would be plotted and calculated between 0 and 1 a_0 only with $\text{xscale}=1$. For **Neon2** xscale is set to 10 and it is necessary to adjust xshift to -0.4 to display all relevant features of the potential in the array.

Note The i -index for the x -axis is not distance/bohr but the array index; use the formula given at the beginning of this section to calculate the distance.

xshift: *default value = 1*

Altering the value of xshift results in shifting the potential in the array boundaries. The harmonic potential, for example, is shifted by $\text{xshift} = 0.5$. Be careful in using this option because shifting important potential features (potential well) beyond the array boundaries results in incorrect energy levels. In addition $\text{xshift} = 0$ might produce problems if the potential goes to infinity at zero.

focus: *default value = 1 or an equivalent: round(max_n_dim/max_n_dim)*

The parameter **focus** works as zoom. If you have optimized xshift and xscale and the potential well is displayed too small, you can zoom in by increasing the value of the focus parameter. Altering the focus parameter has no influence on the calculation and results because it is specific to only the display. If the potential well (e.g. **Neon2**) is not clearly visible, focus according to the following recipe.

- 1) Verify at which array index the left potential flank approximately crosses the x -axis.
- 2) Replace the numerator in the default expression "**round(max_n_dim/max_n_dim)**" in such a way that the term becomes a little smaller than the array index determined in the first step.
- 3) If the zoom effect is too small, use a smaller number for the numerator.

Note that one can insert directly a number for the focus parameter, but then one must always readjust the focus if one increases accuracy on increasing array resolution (max_n_dim).

Other parameters (global)

func_modulator: see advanced utilities

potentialscaling

The parameter **potentialscaling** is a constant factor multiplying the whole potential function. Altering its value influences the potential energy (y -axis). The energy in the output plot is given in microhartree by default for physically meaningful cases (Ne-dimer, H_2 et cetera). If other energy units are appropriate, implementation is effected on adjusting the value of the **potentialscaling** parameter.

globals and arrays

For the input variables see "introduction and initialization" -> "initiating execution".

Array dimension of acquisition array and selection of orbitals being calculated.

```
> numb_orbitals := to_orbital - from_orbital + 1:
   wavefunction_index := 1:
> Acquisition := array(0..(2*numb_orbitals), 0..max_n_dim):
```

This assignment stores the orbitals in Acquisition[1..numb_orbitals,i] and lines with the eigenvalues in Acquisition[num_orbitals+1..2*numb_orbitals,i] for the output graph.

> Potential := 0:

Acquisition[0,i]=Acquisition[Potential,i] stores the potential function numerically.

Epsilon --> Numerov accuracy (error)

> epsilon := 1.0e-8:

This accuracy is expected to be satisfactory for the implemented potentials; for another potential adjustment might be necessary.

program procedures

filling of potential and determination of potential minimum

```
> fillpotential := proc(max_n,Acquisition,potential_min::evaln)
    local infinit, k;
    infinit := 100000;
    potential_min := infinit;
    selected_potential(max_n,Acquisition,focus,mue,xscale,
        potentialscaling,func_modulator,unam); # fills potential
into Acquisition[0,i]
    for k from 1 to max_n do
        if Acquisition[Potential,k] < eval(potential_min) then
            potential_min := Acquisition[Potential,k];
        end if;
    end do; # finds potential minimum
end proc;
```

initial values of energy

```
> startvalues := proc(max_n,numb_of_wished_nodes,to_orbital,E,
    Acquisition,E_lower,E_upper,E_lower_keep,point_min)
    local k, E_upper_start,infinit;
    infinit := 100000;
    if eval(point_min) then # first trial for each
eigenfunction
        E_upper_start := -infinit;
        for k from 1 to max_n do
            if Acquisition[Potential,k] > E_upper_start then
                E_upper_start := Acquisition[Potential,k];
            end if;
        end do;
        if Acquisition[Potential,1] = Acquisition[Potential,2] then
            E_upper_start := infinit;
        end if;
        if (numb_of_wished_nodes = to_orbital - 1) then
            E_lower := potential_min;
            E_upper := E_upper_start;
```



```

        E_lower_keep := eval(E_lower);
    else E_lower := E_lower_keep;
        E_upper := E;
    end if;
else      # if in a first calculation with a smaller array
          # (see speedup above) an eigenvalue E was found that is
used as initial value
    if    eval(E) > 0 then
        E_lower := 0.9 * E;
        E_upper := 1.3 * E;
    else
        E_lower := 1.1 * E;
        E_upper := 0.7 * E;
    end if;
end if;
end proc:

```

Numerov

```

> U := proc(n,prefactor,E,Acquisition,T::evaln)
    local U_return;
    T := eval(prefactor) * (Acquisition[Potential,n] - eval(E));
    U_return := (2 + 10 * eval(T)) / (1 - eval(T));
    U_return;
end proc:

> Numerov := proc(max_n,E,prefactor,nodes::evaln,M,
                  dE::evaln,Acquisition)
    local k,n,T,nodes_internal;
    Acquisition[wavefunction_index,max_n] := 1.0e300;
    Acquisition[wavefunction_index,1]      := 1.0e300;
    k := max_n;
    n := k - 1;
    T := 2;
    while not ((T < 1) and (Acquisition[wavefunction_index,n] <= 1)
or (n = 2)) do
        n := k - 1;                                # iteration from the
right
        Acquisition[wavefunction_index,n] := U(n,prefactor,E,
Acquisition,T)
        - 1.0 / Acquisition[wavefunction_index,k];
        k := k - 1;
    end do;
    M := n;
    nodes_internal := 0;

```

```

    nodes := nodes_internal;
    for k from 2 to (eval(M) - 1) do           # iteration from
the left
        Acquisition[wavefunction_index,k] := U(k,prefactor,E,
Acquisition,T)
            - 1.0 / Acquisition[wavefunction_index,k - 1];
        if (T < 1) and (Acquisition[wavefunction_index,k] < 0) then
            nodes_internal := nodes_internal + 1;
            nodes := nodes_internal;
        end if;
    end do;
    dE := 1.0 / Acquisition[wavefunction_index,eval(M)+1] -
        U(M,prefactor,E,Acquisition,T) + 1.0 /
        Acquisition[wavefunction_index,eval(M)-1];
end proc:
secant & bisecant method
> secant_method := proc(dE,dEold,E::evaln,Eold::evaln)
    local Enew,E_numerator,E_denominator;
    E_numerator := eval(E) * dEold - eval(Eold) * dE;
    E_denominator := dEold - dE;
    if E_denominator = 0 then
        Enew := eval(E);
    else
        Enew := E_numerator/E_denominator;
    end if;
    Eold := eval(E);
    E := Enew;
end proc:
> bisect := proc(nodes,numb_of_wished_nodes,dE,nodeequal::evaln,
    E::evaln,E_lower::evaln,E_upper::evaln,Eold::evaln)
    if (nodes > numb_of_wished_nodes) then
        E_upper := eval(E);
    elif (nodes < numb_of_wished_nodes) then
        E_lower := eval(E);
    else
        if dE < 0 then
            E_lower := eval(E);
        else
            E_upper := eval(E);
        end if;
        if (abs(dE) < 1.0) then
            nodeequal := true;

```

```

        end if;
    end if;
    Eold := eval(E);
    E := 0.5 * (eval(E_lower) + eval(E_upper));
end proc;
find eigenvalue
> FindEigenvalue := proc(max_n,numb_of_wished_nodes,to_orbital,
    point_min,E_lower::evaln,E_lower_keep,E_upper::evaln,
    prefactor::evaln,E::evaln,epsilon,M::evaln,
    Acquisition,iteration::evaln)
    local dE,dEold,Eold,nodeequal,nodes,conv_criterium,
        Enew,iteration_number;
    prefactor := evalf(mue /(6 * potentialscaling * ((max_n - 1) /
        xscale)^2)):
    startvalues(max_n,numb_of_wished_nodes,to_orbital,E,
        Acquisition,E_lower,E_upper,E_lower_keep,point_min);
    nodeequal      := false;
    iteration_number := 1;
    conv_criterium  := 1;
    Eold            := 20000;
    Enew            := 0.5 * (eval(E_lower) + eval(E_upper));
    while epsilon < conv_criterium do
        dEold := dE;
        conv_criterium := abs((Enew - Eold) / Enew);
        Numerov(max_n,Enew,prefactor,nodes,M,dE,Acquisition);
        if (nodeequal) and (nodes = numb_of_wished_nodes) then
            secant_method(dE,dEold,Enew,Eold);
        else bisect(nodes,numb_of_wished_nodes,dE,nodeequal,Enew,
            E_lower,E_upper,Eold);
        end if;
        iteration_number := iteration_number + 1;
        iteration        := iteration_number;
    end do;
    E := Enew;
end proc;
find eigenfunction
> FindEigenfunction := proc(max_n_dim,M,prefactor,E,Acquisition,
    wavefunction_index)
    local k,T;
    Acquisition[wavefunction_index,M] := 1;
    for k from (M - 1) to 1 by -1 do
        Acquisition[wavefunction_index,k] :=

```

```

        Acquisition[wavefunction_index,k + 1] /
        Acquisition[wavefunction_index,k];
    end do;
    for k from (M + 1) to max_n_dim by 1 do
        Acquisition[wavefunction_index,k] :=
            Acquisition[wavefunction_index,k - 1] /
            Acquisition[wavefunction_index,k];
    end do;
    for k from 1 to max_n_dim by 1 do
        T := prefactor * (Acquisition[Potential,k] - E);
        Acquisition[wavefunction_index,k] :=
            Acquisition[wavefunction_index,k] / (1 - T);
    end do;
end proc:
acquisition array -> energy storage and wavefunction modulation
> Acquisition_modulation := proc(Acquisition,E)
    local k;
    Acquisition[wavefunction_index,0] := E;
    for k from 1 to max_n_dim do
        Acquisition[wavefunction_index,k] := func_modulator *
            Acquisition[wavefunction_index,k] +
            Acquisition[wavefunction_index,0];
        Acquisition[(wavefunction_index + numb_orbitals),k] :=
            Acquisition[wavefunction_index,0];
    end do;
end proc:
plotting
> with(plots):
with(plottools):
output_plot := proc(Acquisition,numb_orbitals,focus)
    local j,color,Acquisition_color;
    color[0]:= red:
    color[1]:= blue:
    color[2]:= coral:
    color[3]:= green:
    Acquisition_color := array(0..(2*numb_orbitals));
    Acquisition_color[Potential] :=
        curve([(seq([k,Acquisition[Potential,k]],
            k=focus..max_n_dim))],colour=black,style=LINE, thickness=4):
    for j from 1 to numb_orbitals do
        Acquisition_color[j] :=
            curve([(seq([k,Acquisition[j,k]],k=1..max_n_dim))],

```

```

        colour=color[j mod 4],style=LINE, thickness=2):
end do:
for j from (numb_orbitals + 1) to (2*numb_orbitals) do
    Acquisition_color[j] :=
        pointplot({seq([k,Acquisition[j,k]],k=1..max_n_dim)},
            colour=black,symbol=point):
end do:
display([seq(Acquisition_color[i],i=0..(2*numb_orbitals))],
    title="display of potential and wave functions");

# ,
#     view=[0..2200, -120..20]
# To limit your graph, you might use as option: view=[x1..
x2,y1..y2]
end proc:

```

Warning, (in output_plot) `k` is implicitly declared local
Warning, (in output_plot) `i` is implicitly declared local

To limit your graph, you might use as option: `view=[x1..x2,y1..y2]`.

main program

```

> numb_of_wished_nodes := to_orbital - 1:
> E := 10000:
if from_orbital = 0 or to_orbital = 0 then
    WARNING("Orbital 0 is not a valid choice! Please interrupt the
calculation
    by pressing the STOP button in the menu and initiate a new
calculation using legitimate orbital numbers.");
end if;
while numb_of_wished_nodes <> from_orbital - 2 do
    max_n := max_n_dim/speedup/10; # loop over desired orbitals
(eigenfunctions) starting with the energetically highest
    point_min := true;
    while max_n < max_n_dim do
        max_n := max_n*10; # loop over differently sized
Acquisition arrays
                                # (normally one with size max_n_dim/10
and max_n_dim)
                                # to accelerate the calculation (see
speedup above)
        fillpotential(max_n,Acquisition,potential_min);
        FindEigenvalue(max_n,numb_of_wished_nodes,to_orbital,
point_min,
            E_lower,E_lower_keep,E_upper,prefactor,E,epsilon,M,
Acquisition, iteration);
        point_min := false;

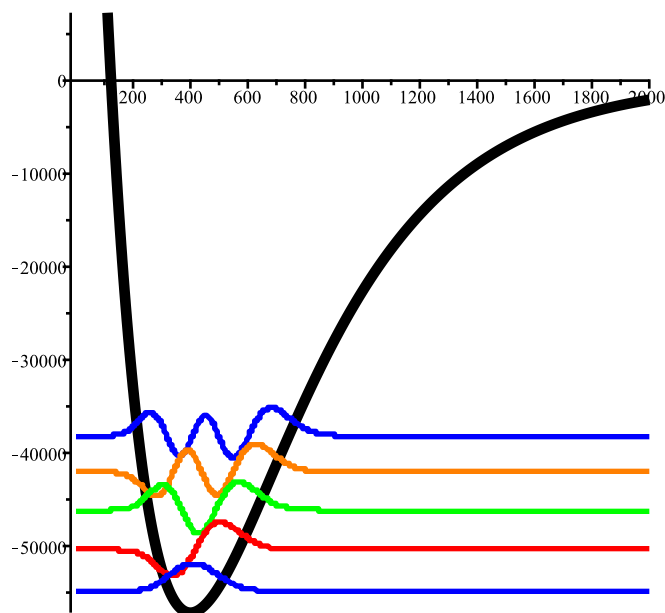
```

```

end do;                                # end of loop over the differently
sized Acquisition arrays
FindEigenfunction(max_n_dim,M,prefactor,E,Acquisition,
    wavefunction_index):
Acquisition_modulation(Acquisition,E);
print(Orbital,numb_of_wished_nodes + 1,"energy /", unam,
    Acquisition[wavefunction_index,0],"number of iterations =",
    (iteration));
numb_of_wished_nodes := numb_of_wished_nodes - 1;
wavefunction_index := wavefunction_index + 1;
end do:                                # end of loop over the orbitals
output_plot(Acquisition,numb_orbitals,focus);
Orbital, 5, "energy /", "1/cm", -38178.1064749055436277408, "number of iterations =", 10
Orbital, 4, "energy /", "1/cm", -42066.3954109718209253563, "number of iterations =", 13
Orbital, 3, "energy /", "1/cm", -46143.2041334539540164588, "number of iterations =", 14
Orbital, 2, "energy /", "1/cm", -50408.5326418762116843024, "number of iterations =", 14
Orbital, 1, "energy /", "1/cm", -54862.3809349249170476052, "number of iterations =", 15

```

display of potential and wave functions



e3.71 exercise

Add another example to the list of selectable functions for potential energy in the list in the preceding section.

3.72 numerical solution of Schroedinger's equation for potential energy II

```

> restart:
Digits := 32:

```

The following procedures for the numerical solution of Schroedinger's equation for a formula for potential energy in one dimension and some examples of its use are adapted from programmes in Basic and ancillary material kindly provided by Professor Joel Tellinghuisen, Vanderbilt University [*Journal of Chemical Education*, **66** (1989) 51] to solve Schroedinger's equation in a form

$$\frac{\partial^2}{\partial r^2} \psi = \left(\frac{8 \pi^2 \mu (U(r) - E)}{h^2} + \frac{J(J+1)}{r^2} \right) \psi$$

in which a term containing J pertains to angular momentum, also according to Numerov's method. The boundary condition is that $\psi(r) \rightarrow 0$ as $r \rightarrow 0$.

```
> Eigfun := proc(Uu,mu,N,R1,R2,J1,En)
  local C,DJ,E,F,G,H,J0,NV,N0,P,Pe,Pp,Q,S,S1,S2,T,U,V,W,X,Y0,Y1,
  Y2,Y3,i,j,k;
  if R1 = 0 then
    N0 := 1;
  else      # for H atom
    N0 := 0;      # otherwise
  end if;
  C := 0.05932032247844*mu;
  S := (R2 - R1)/N :
  W := S^2*C :
  T := W/12 :
  J0 := 0 :
  E := En;
  for i from 0 to N do
    X := R1 + S*i:
    U[i] := evalf(eval(Uu, R=X));
  end do:
  DJ := (J1^2 + J1 - J0^2 - J0)/C:
  for i from 0 to N do
    X := R1 + S*i:
    U[i] := U[i] + DJ/X^2
  end do:
  # Start inward integration from right end.
  35: P[N] := 1.0e-18:
  P[N-1] := 2.0e-18:
  V := U[N] - E:
  if V<0 then
    ERROR(`E>U at right end:  increase R2.`);
  end if;
  Y2 := P[N]*(1 - T*V);
  V := U[N-1] - E;
  Y3 := P[N-1]*(1 - T*V);
  for i from 2 to N do
```

```

        j := N - i;
        Y1 := Y2;
        Y2 := Y3;
        Y3 := 2*Y2 - Y1 + W*V*P[j+1];
        V := U[j] - E;
        P[j] := Y3/(1 - T*V);
#   print(`J=`, j, `  Y=`, Y3, `  P=`, P[j]);
        if P[j] < P[j+1] then
            goto(80);
#   When psi begins to decrease, escape from loop and rescale psi
    so that
#   outgoing psi matches ingoing psi[k]=1, with k as index for
    which P[k]<P[k+1].
        end if;
    end do;
80: k := j;
    Q := 1/P[k];
    Y0 := Y2*Q;
    for i from k to N do
        P[i] := P[i]*Q;
    end do;
#   Begin outward integration, from i=0 to i=k; then rescale psi so
    that
#   outgoing psi matches ingoing psi at i=k; i.e. P[k] = 1.
    P[0] := 1.0e-18;
    P[1] := 2.0e-18;
    V := U[0] - E;
    if V<0 then
        error(`E > U at left end.  Decrease R1.`)
    end if;
    Y2 := P[0]*(1 - T*V);
    V := U[1] - E;
    Y3 := P[1]*(1 - T*V);
    for i from 2 to k do
        Y1 := Y2;
        Y2 := Y3;
        Y3 := 2*Y2 - Y1 + W*V*P[i-1];
        V := U[i] - E;
        P[i] := Y3/(1 - T*V);
#   print(`i=`, i, `  Y=`, Y3, `  P=`, P[i]);
    end do;
    Q := 1/P[k];

```



```

Y2 := Y2*Q;
Y3 := Y3*Q;
for i from 0 to k do
    P[i] := P[i]*Q;
end do;
# Calculate correction to E and return for another cycle; repeat
until dE < 0.003.
G := 0;
for i from 0 to N do
    G := G + P[i]*P[i];
end do;
F := (2*Y3 - Y2 - Y0)/S^2 + C*V;
F := F/G/C;
E := E + F;
print(`dE = `, F);
if abs(F) > 0.003 then
    goto(35);
end if;
# Count nodes to evaluate v.
NV := 0;
S1 := sign(P[0]);
for i from 1 to N do
    S2 := sign(P[i]);
    if S2<>S1 then
        NV := NV + 1;
        S1 := S2;
    end if;
end do;
Q := 1/sqrt(S*G);
# Normalise psi and calculate B_v that is proportional to
<1/R^2>.
H := 0;
for i from 0 to N do
    X := R1 + S*i;
    Pe := P[i]*Q;
    Pp := Pe/X;
    H := H + Pp^2;
    P[i] := Pe;
end do;
H := H*S;
print(`v = `, NV, ` J = `, J1, ` E(v,J) = `, E); #, `Bv = `,
H/C);

```

```

    for i from N0 to N do
    #   print(`i = `, i,` P[i] = `, P[i]);
    end do;
end proc:

```

As a first example of its use we try Morse's function for potential energy:

$U(R) = D_e \left(1 - e^{-\alpha(R - R_e)} \right)^2$ that serves as a model of internuclear potential energy of a diatomic molecule. We take equilibrium binding energy $D_e / \text{cm}^{-1} = 30000$, $\alpha = \frac{a_M}{R_e} = 1.0$ with a_M being a parameter characterizing Morse's function, equilibrium internuclear distance $R_e = 1.5$ in unit 10^{-10}m , and reduced mass $\mu = 10 \text{ u}$, for which $B_e = 0.7492$ in the same units as D_e and $U(R)$. Here are calculated B_e ,

```

> Be := evalf(6.62606876e-34*6.02214199e26/(8*Pi^2*299792458*10*
(1.5e-10)^2))/100;
Be := 0.74922795966987044875379539005445

```

(13.1)

and a formula corresponding to Morse's function with imposed values of parameters.

```

> Uu := 30000*(1-exp(-1.0*(R-1.5)))^2;
Uu := 30000 (1 - e-1.0 R + 1.50)2

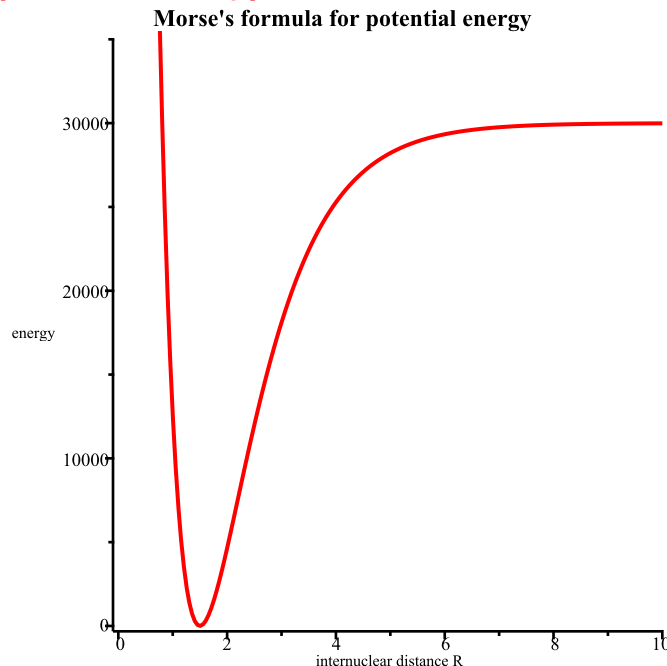
```

(13.2)

```

> plot(Uu, R=0..10,-3..35000,labels=["internuclear distance R",
"energy"], axes=frame,
title="Morse's formula for potential energy", titlefont=
[TIMES,BOLD,14], colour=red);

```



```

> UuAr := .12522237e9*exp(-3.8514698*R)-37022.895*exp(-1.4532943*R)
+ 92.793874; # argon Woon pm/100, cm-1, 0 at minimum

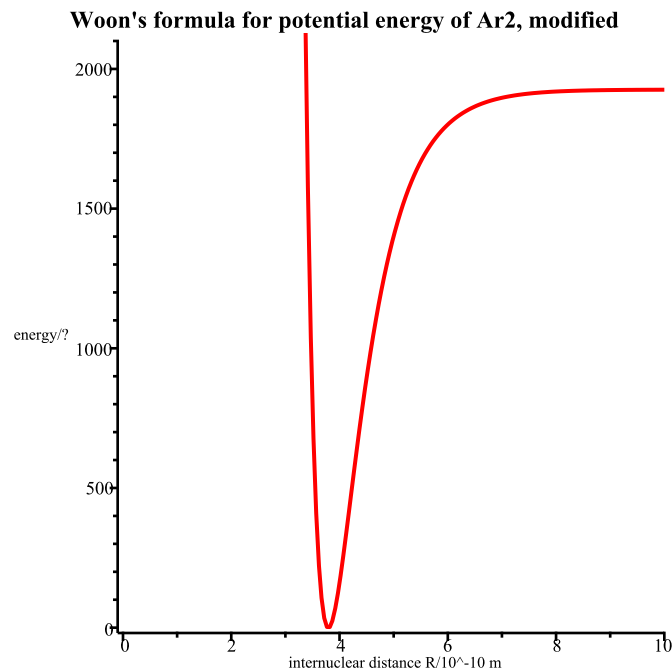
```

$$UuAr := 1.2522237 \times 10^8 e^{-3.8514698 R} - 37022.895 e^{-1.4532943 R} + 92.793874 \quad (13.3)$$

$$\begin{aligned} &> Uu2 := 1000 * (570555.1779 * \exp(-2.038109223 * R / 0.529) - 168.6887442 * \exp \\ &\quad (-.7690499311 * R / 0.529)) / (2 * 0.1096775836756712) + 1926; \\ Uu2 &:= 2.6010564728850840166487605800915 \times 10^9 e^{-3.8527584555765595463137996219282 R} \end{aligned} \quad (13.4)$$

$$- 769021.06404363975482899795929114 e^{-1.4537805880907372400756143667297 R} + 1926$$

```
> plot(Uu2, R=0..10,-0..2100,labels=["internuclear distance R/10^-10
m","energy/?"],
      axes=frame, titlefont=[TIMES,BOLD,14], colour=red,
      title="Woon's formula for potential energy of Ar2,
modified");
```



In the call to this procedure, **Eigfun**($Uu, \mu, N, R1, R2, J, E_{est}$), arguments are

- a formula for a function for potential energy Uu ,
- reduced mass μ/u ,
- number N of steps in numerical integration,
- left $R1$ and
- right $R2$ bounds of integration/ 10^{-10} m,
- quantum number J for rotational energy, and
- an initial estimate of energy in the same units as Uu of a bound state.

Too small initial estimates of an energy for $v=0$ cause unpredictable results: if a demand to increase $r2$ occur, increase the initial estimate instead.

```
> #Eigfun(Uu2,36423.46,1000,0.6,5.5,0,2000.);
```

```
#Eigfun(Uu,1,1000,0.5,5.5,0,1000.):
> Eigfun(UuAr,19.974,1000,2.5,8,0,20.):
    dE = , -4.5570097893354119630705333809397
    dE = , -1.2169716599990452032836717301762
    dE = , -0.043358434042817385395786600622553
    dE = , -0.000053172666475997588747165164971721
    v = , 0, J = , 0, E(v,J) = , 14.182606943956249450661261123096
```

(13.5)

```
> Eigfun(UuAr,19.974,1000,2.5,8,0,30.):
    dE = , 6.4738146717964470709475990830516
    dE = , 2.3657674005237224098192841019092
    dE = , -0.10200815413780397121698103917511
    dE = , -0.00030142212846434306233540319939550
    v = , 1, J = , 0, E(v,J) = , 38.737272496053901166487566742587
```

(13.6)

```
> Eigfun(UuAr,19.974,1000,2.5,8,0,50.):
    dE = , 3.3648522790460580869743237929091
    dE = , 4.8668915727613222814537802304064
    dE = , 0.17478369624985700652595433523606
    dE = , -0.0010615880842637603756567573990675
    v = , 2, J = , 0, E(v,J) = , 58.405465959972973614578401601152
```

(13.7)

```
> Eigfun(UuAr,19.974,1000,2.5,8,0,70.):
    dE = , 3.4276915021874449179342111743296
    dE = , -0.036160153330230170039150874519062
    dE = , -0.000070267428608393657898658555543813
    v = , 3, J = , 0, E(v,J) = , 73.391461081428606354237161641255
```

(13.8)

```
> Eigfun(UuAr,19.974,1000,2.5,8,0,80.):
    dE = , 2.6071066420451294624044812521405
    dE = , 1.4169813136635341178713630455893
    dE = , -0.092702363976343265283352992038268
    dE = , -0.00075309798645723095690617936912791
    v = , 4, J = , 0, E(v,J) = , 83.930632493745863084035585126322
```

(13.9)

```
> Eigfun(UuAr,19.974,1000,2.5,8,0,90.):
    dE = , 0.31991168458702764312407489157048
    dE = , -0.014416523546697803254629415443872
    dE = , -0.000035727345119367993523679645481742
```

$$\nu = , 5, \quad J = , 0, \quad E(\nu, J) = , 90.305459433695210471875921796481 \quad (13.10)$$

The differences between these vibrational terms are

```
> 38.737272496053901166487566742587
-14.182606943956249450661261123096;
58.405465959972973614578401601152
-38.737272496053901166487566742587;
73.391461081428606354237161641255
-58.405465959972973614578401601152;
83.930632493745863084035585126322
-73.391461081428606354237161641255;
90.305459433695210471875921796481
-83.930632493745863084035585126322;
24.554665552097651715826305619491
19.668193463919072448090834858565
14.985995121455632739658760040103
10.539171412317256729798423485067
6.374826939949347387840336670159 (13.11)
```

One can make this procedure effectively semi-automatic by merely giving it, in a single or double **for** loop, a formula for potential energy and a range of vibrational or rotational states for which the energy is calculated. Energy is not used directly, but wavenumber $\bar{\nu}$, which is energy divided by Planck's constant h and speed c of light in vacuo. By differentiating the given formula for potential energy in a common case of a molecule, one can calculate the equilibrium internuclear distance R_e , which is the distance for which the potential energy is zero at the minimum of the curve denoting the formula: $\frac{d}{dR} V(R) = 0$ at $R = R_e$. With vibrational quantum number ν and rotational quantum number

J , the state of least energy, with $\nu = J = 0$, occurs at $\frac{\omega_e}{2}$, and a reasonable initial estimate of energies of

states of greater vibrational energy is $\frac{\omega_2}{2} + 0.97^\nu \omega_e$ or a similar formula, whereas an initial estimate

of energies of states of greater rotational energy is $B J (J + 1)$ in which $B = \frac{h}{8 \pi^2 c \mu R_e^2} =$

$\frac{1.6857629 \cdot 10^{-16}}{\mu \left(\frac{R_e}{m} \right)^2}$, which assume that μ is expressed in unified atomic mass unit u and R_e in m.

```
> for v from 0 to 10 do
  for J from 0 to 1 do
    Eigfun(Uu, 10, 1000, 0.7, 5.5, J, 460.00*(v+1/2)*0.99^v);
  end do;
```

end do:

```
dE = , -5.4879096494540912567383039658971
dE = , -0.049366337878991095531272641204227
dE = , -0.000016472323798389154423850553089019
v = , 0, J = , 0, E(v,J) = , 224.46270754034311925857599954235
dE = , -4.0160548186553642153970960789006
dE = , -0.026551353915047976617321956118549
dE = , -7.8968357034297549294110194882933 × 10-6
v = , 0, J = , 1, E(v,J) = , 225.95738593059388437823065255396
dE = , -12.062812235849210236488720019612
dE = , -0.17765225113127884401326715297549
dE = , -0.00012130776669411272281232214942650
v = , 1, J = , 0, E(v,J) = , 670.85941420525281680677520050526
dE = , -10.617059328144719944975163128922
dE = , -0.13629939641384931269996096969322
dE = , -0.000087115619185511007768720381406726
v = , 1, J = , 1, E(v,J) = , 672.34655415982224523131710718101
dE = , -13.029505384377784123278139118077
dE = , -0.20092749351043416497958417371436
dE = , -0.00016016770387475120308202858543193
v = , 2, J = , 0, E(v,J) = , 1113.8844069544079069605391946796
dE = , -11.593974412720569934645144710806
dE = , -0.15694856043751970284939004163995
dE = , -0.00011838684395499114462612092419874
v = , 2, J = , 1, E(v,J) = , 1115.3639586399979553713608391267
dE = , -8.5560386096270219082032269660675
dE = , -0.087715488115742823643683143489282
dE = , -0.000065772132722054739358741873826552
v = , 3, J = , 0, E(v,J) = , 1553.5375701301245132134137311485
dE = , -7.1117005450528937294991751031624
dE = , -0.060162219866788326544006224662396
dE = , -0.000043469688132587664771139755721447
v = , 3, J = , 1, E(v,J) = , 1555.0094837653921853562920475323
```

$$\begin{aligned}
dE &= , 1.3853946076314063625514344726697 \\
dE &= , -0.00033854638177405225890232556147713 \\
v &= , 4, \quad J = , 0, \quad E(v,J) = , 1989.8187967612496323102925321471 \\
dE &= , 2.8527793599489823012005653452841 \\
dE &= , -0.0034950981249946890167146925805086 \\
dE &= , -2.6200470972490034085774639444486 \times 10^{-6} \\
v &= , 4, \quad J = , 1, \quad E(v,J) = , 1991.2830223417768903631804420752 \\
dE &= , 16.850965875107634978371692016607 \\
dE &= , -0.12769314513884662571555897079781 \\
dE &= , -0.00011205216737201949175873284574721 \\
v &= , 5, \quad J = , 0, \quad E(v,J) = , 2422.7279869248014163331643743130 \\
dE &= , 18.323565605052000733823489729513 \\
dE &= , -0.14378868758881273993258508062870 \\
dE &= , -0.00012792719088968443000567581246017 \\
v &= , 5, \quad J = , 1, \quad E(v,J) = , 2424.1844752372722983094608989731 \\
dE &= , 37.171697089556052298310315180506 \\
dE &= , 0.067658181417285968302854586891540 \\
dE &= , 0.000047582343176854315987925210824022 \\
v &= , 6, \quad J = , 0, \quad E(v,J) = , 2852.2650495623065151209291576926 \\
dE &= , 38.574012389349291505050595642653 \\
dE &= , 0.11401662245970342291453576883422 \\
dE &= , 0.000074922255552658797711691411209166 \\
v &= , 6, \quad J = , 1, \quad E(v,J) = , 2853.7137506430545475867628431029 \\
dE &= , 58.738891995075765418137192151048 \\
dE &= , 4.0704444793411295067777062785051 \\
dE &= , -0.0048819461330827823387080259560709 \\
dE &= , -3.8973966596851247656918904098260 \times 10^{-6} \\
v &= , 7, \quad J = , 0, \quad E(v,J) = , 3278.4299009100026524574514247116 \\
dE &= , 59.858932683668678358781852440085 \\
dE &= , 4.3921962850670067105890685806870 \\
dE &= , -0.0058094046945107266339527152467779 \\
dE &= , -4.6462586426214494557083107261425 \times 10^{-6}
\end{aligned}$$

$v = , 7, J = , 1, E(v,J) = , 3279.8707651968980317212875125973$
 $dE = , 79.721658009548175753828840088755$
 $dE = , 13.647068067392495103490601550178$
 $dE = , -0.077945660125601135158432499498776$
 $dE = , -0.000071328714309540237052841659302293$
 $v = , 8, J = , 0, E(v,J) = , 3701.2224643012683511819239562978$
 $dE = , 80.428691475130990719583290953366$
 $dE = , 14.378871377849210744768402430420$
 $dE = , -0.083798796783517892751797172777696$
 $dE = , -0.000077058801619477596451831689239628$
 $v = , 8, J = , 1, E(v,J) = , 3702.6554422105626550940034443793$
 $dE = , 85.252700209100184484440242782747$
 $dE = , 42.729587423934920962070175597397$
 $dE = , 0.58986592864526632591126106465232$
 $dE = , 0.00014503828932181468808265223428474$
 $v = , 9, J = , 0, E(v,J) = , 4120.6426701034804222171097620970$
 $dE = , 85.140742587753007201748621407852$
 $dE = , 44.156569165491692042797440296369$
 $dE = , 0.69993005819965217617390500938687$
 $dE = , 0.000098836263922805199204213076802349$
 $v = , 9, J = , 1, E(v,J) = , 4122.0677121512190028559191709268$
 $dE = , 51.515164358760148495796912991176$
 $dE = , 81.456682307111586724717932386207$
 $dE = , 35.159138423187574723564475951382$
 $dE = , 0.39384806381036453701011359656199$
 $dE = , 0.00020079782315574562518595905764825$
 $v = , 10, J = , 0, E(v,J) = , 4536.6904562432185169750146208845$
 $dE = , 50.297982830364347345455444327500$
 $dE = , 81.595940080999437780857483565350$
 $dE = , 37.520893432059414336951209998583$
 $dE = , 0.52706610017146284306077728414026$
 $dE = , 0.00020815720460196660922157396295708$

$$v = , 10, \quad J = , 1, \quad E(v,J) = , 4538.1075128933249510212341367495 \quad (13.12)$$

The output indicates the vibrational quantum number of the nearest state to the initial estimate, the same rotational quantum number as input, and an accurate energy of that quantum state. For Morse's function for potential energy, vibrational eigenvalues for $J=0$ are calculated according to a formula for vibrational terms:

$$E_M = \omega_e \left(v + \frac{1}{2} \right) - a_M^2 B_e \left(v + \frac{1}{2} \right)^2.$$

```
> Em := v -> 449.76834*(v+1/2) - 1.6857630*(v+1/2)^2;
```

$$Em := v \mapsto 449.76834 \cdot \left(v + \frac{1}{2}\right) - 1.6857630 \cdot \left(v + \frac{1}{2}\right)^2 \quad (13.13)$$

```
> seq(evalf(Em(v), 12), v=0..10);
```

224.462729250, 670.859543250, 1113.88483125, 1553.53859325, 1989.82082925, **(13.14)**
2422.73153925, 2852.27072325, 3278.43838125, 3701.23451325, 4120.65911925,
4536.71219925

As another test we try these data that might be applicable to O_7 .

```
> V := 56856.60*(1-exp(-3.2*(R-1.207)/1.207))^2;
```

```
for v from 0 to 1 do
```

```
for J from 0 to 1 do
```

```
Eigfun(V, 8, 2000, 0.7, 5.5, J, 1600.00*(v+1/2));
```

end do:

end do:

$$V := 56856.60 \left(1 - e^{-2.6512013256006628003314001657001 R + 3.2000000000000000000000000000} \right)^2$$

$$dE = , 119.23960061130075973034088540103$$

$$dE = , -5.2609390670994506888218569274356$$

$$dE = , -0.011789357537230487736334262035329$$

$$dE = , -2.5039007793773067842022568751253 \times 10^{-6}$$

$$v = , 0, \quad J = , 0, \quad E(v,J) = , 913.96686968276329917647591000929$$

$$dE = , 122.38270408099424089952950601964$$

$$dE = , -5.5251398745235485943102811256973$$

$$dE = , -0.012949425756164589533229933194282$$

$$dE = , -2.7560410721559975852648427720984 \times 10^{-6}$$

$$v = , 0, \quad J = , 1, \quad E(v,J) = , 916.8446120246734555968840969591$$

$$dE = ,316.25640227027900307163904199136$$

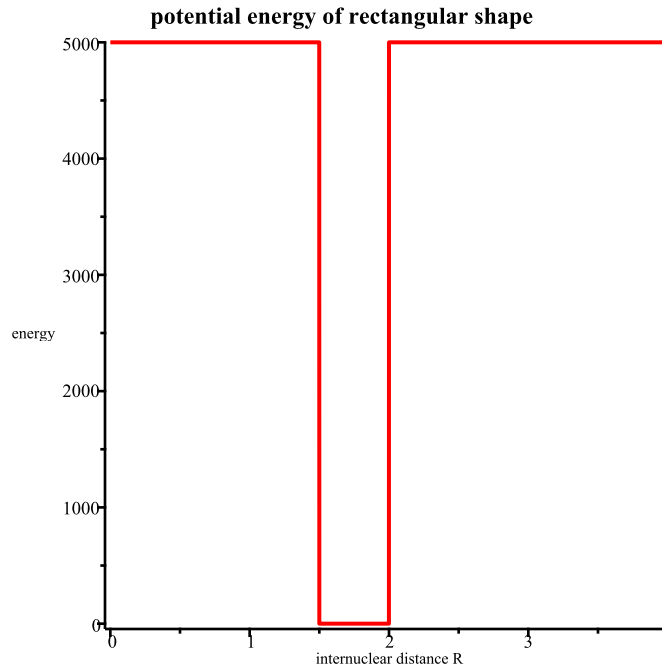
$$dE = , 3.4290833232144534945363942078122$$

$$dE = , -0.0019286551226809491642858229920287$$

$$\begin{aligned}
v &= , 1, \quad J = , 0, \quad E(v,J) = , 2719.6835569383707756170111503762 \\
dE &= , 318.80509775640056418435493595283 \\
dE &= , 3.7283701980207367756528746105726 \\
dE &= , -0.0024007646797056068940826174359836 \\
v &= , 1, \quad J = , 1, \quad E(v,J) = , 2722.5310671897415953531137279460
\end{aligned} \tag{13.15}$$

The next example is a formula that plots to a rectangular shape according to which energy is 0 in a range/ 10^{-10} m [1.5, 2] and 5000 units elsewhere; we achieve this formula through use of Heaviside functions.

$$\begin{aligned}
> \text{Uu} &:= 5000 * (\text{Heaviside}(R-2) - \text{Heaviside}(R-1.5) + 1); \\
\text{Uu} &:= 5000 \text{Heaviside}(R-2) - 5000 \text{Heaviside}(R-1.5) + 5000 \\
> \text{plot}(\text{Uu}, R=0..4, \text{labels}=["\text{internuclear distance } R", "\text{energy}"], \text{axes}= \\
&\text{frame, colour=red,} \\
&\text{title}=\text{"potential energy of rectangular shape", titlefont=} \\
&[\text{TIMES, BOLD, 14}]);
\end{aligned} \tag{13.16}$$



$$\begin{aligned}
> \text{Eigfun}(\text{Uu}, 10, 1000, 1.35, 2.15, 0, 4500.): \\
dE &= , -14.323853819428976997933701508246 \\
dE &= , -0.033834539592754931507408572166239 \\
dE &= , -4.8796392178621360048269262055792 \times 10^{-6} \\
v &= , 8, \quad J = , 0, \quad E(v,J) = , 4485.6423067613390502084228850927
\end{aligned} \tag{13.17}$$

For a formula potential energy that generates this shape, exact eigenvalues are solved as roots of these equations:

$$\alpha \tan(\alpha \beta) = \sqrt{1 - \alpha^2} \quad \text{for even solutions}$$

$\alpha \cot(\alpha \beta) = -\sqrt{1 - \alpha^2}$ for odd solutions
 in which $\alpha^2 = \frac{E}{U_0}$ and $\beta^2 = \frac{2 \pi^2 m U_0 \alpha^2}{h^2}$, with width $\alpha = 0.5$ units and depth U_0 units of the rectangular form, for which $\beta = 13.615285$.

```

> beta := 13.615285;
    fsolve(alpha*tan(alpha*beta) = sqrt(1 - alpha^2), alpha=0.9);
        beta := 13.615285
        0.94698371101021922331300458458019
  
```

(13.18)

```

> E := %^2*5000;
        E := 4483.8907445934319851653901232832
  
```

(13.19)

```

> Eigfun(Uu,10,1000,1.35,2.15,0,3500.):
    dE = , 92.421440321528927741892961221054
    dE = , 7.7344716930884250524932433782833
    dE = , 0.0081913920473882581175611988256446
    dE = , 9.3751020088382506350273681090454 × 10-7
    v = , 7, J = , 0, E(v,J) = , 3600.1641043441749419363288293009
  
```

(13.20)

```

> fsolve(alpha*cot(alpha*beta) = - sqrt(1 - alpha^2), alpha=0.9);
    0.84854237093630438966278590185873
  
```

(13.21)

```

> E := %^2*5000;
        E := 3600.1207763710239628419680501670
  
```

(13.22)

For atom H we take $U(R) = -\frac{1.16140973 \cdot 10^5}{R}$ units of energy and the applicable mass is that of an electron, $m_e = 5.4857990946 \cdot 10^{-4}$ u.

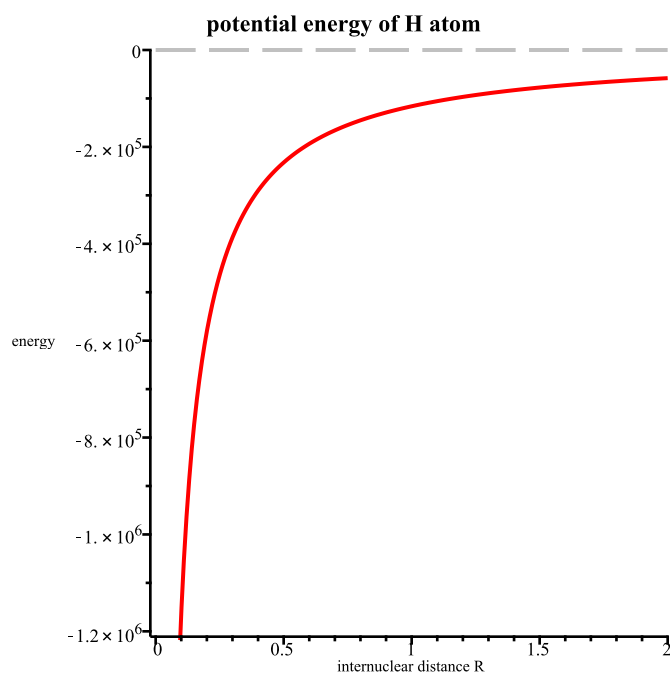
```

> Uu := -0.11614097271111113e6/R;
        Uu := -  $\frac{116140.9727111113}{R}$ 
  
```

(13.23)

```

> plot([Uu, 0],R=0..2, -1.2*10^6..0, labels=["internuclear distance
R","energy"],
    axes=frame, title="potential energy of H atom", titlefont=
[TIMES,BOLD,14],
    colour=[red,greyscale], linestyle=[1,3]);
  
```



```
> Eigfun(Uu,0.00054857990946,512,-0.00003,5,0,-100000.);
      dE = , -9204.7074933473518120311921525426
      dE = , -525.70859685422052894328737542372
      dE = , -1.5760882245813265796556482720485
      dE = , -0.000068759961368344323161943498534625
      v = , 0, J = , 0, E(v,J) = , -109731.99224718611503589845833817
```

(13.24)

```
> Eigfun(Uu,0.00054857990946,750,-0.00003,13,1,-30000.):
      dE = , 2820.9298324439937644230621348296
      dE = , -252.52338205312539180869272481721
      dE = , -2.2765970753857987581697099826703
      dE = , -0.00020901714776215778159327250216658
      v = , 0, J = , 1, E(v,J) = , -27433.870355701665188301581893243
```

(13.25)

```
> Eigfun(Uu,0.00054857990946,512,-0.00003,25,2,-12000.):
      dE = , -189.01696343186698645551062538634
      dE = , -4.0149465619133070293192314894286
      dE = , -0.0018470702731396915387547111731324
      v = , 0, J = , 2, E(v,J) = , -12193.033757064053433176368611586
```

(13.26)

```
> with(ScientificConstants):
      Ri := GetValue(Constant(R[infinity]));
      evalf([-Ri/100, -Ri/400, -Ri/900], 12);
      Ri := 1.0973731568508 x 107
```

The current best value of the Rydberg constant is $(10973731.568539 \pm 0.000055) \text{ m}^{-1}$. The appropriate effective mass for a hydrogen atom comprising a proton and an electron is the reduced mass,

$\mu = \frac{m_e m_p}{m_e + m_p}$, but for comparison with the fundamental physical constant R_∞ we apply the electronic

rest mass m_e in the above calculations. A comparison of results shows that, for $J=0$, the calculated result is near the theoretical result, but for $J=1$ and $J=2$ agreement is nearly exact, with suitably chosen number of points and bounds of integration. When a stable convergence is achieved, decreasing the lower limit and increasing the upper should not alter the result, as long as the number of points sampling the important classical region remains adequate.

Procedure **Eigfun** might perform less well for atom H than for other input functions of potential energy because both the potential energy and amplitude functions vary rapidly near $r=0$ but slowly at large r : a mesh for integration that consists of equally spaced points on r is inefficient. Logarithmic transformation of variable r , as $r = e^x$, yields much improvement, leading to a differential equation

$$\frac{\partial^2}{\partial x^2} Q = \left(C (U(x) - E) e^{2x} + \left(l + \frac{1}{2} \right)^2 \right) Q$$

in which $U(x)$ is a potential energy of coulombic form evaluated at $r = e^x$ and l represents orbital

angular momentum. A relation between Q and $R(r) = \frac{\Psi(r)}{r}$ is $Q = \sqrt{r} R(r)$; the range of the

transformed variable is $-\infty \leq x \leq \infty$. A modified version **EigfunH** of **Eigfun** incorporates these adaptations.

```
> EigfunH := proc(N,R1,R2,J1,En)
    local C,DJ,E,F,G,H,J0,NV,P,Pe,Q,S,S1,S2,T,U,V,W,X,Y0,Y1,Y2,Y3,i,
    j,k,mu,
            Uu,RR,EX,TT,PT,PD;
    mu := 5.4857990946e-4:
    C := 0.0593203224784*mu:
    S := (R2 - R1)/N :
    W := S^2*C :
    T := W/12 :
    E := En:
    for i from 0 to N do
        X := R1 + S*i:
        RR := evalf(exp(X));
        U[i] := -116140.9734/RR;
        EX[i] := RR^2;
    end do:
    DJ := (J1 + 1/2)^2/C:
    # Start inward integration from right end.
    35: P[N] := 1.0e-18:
        P[N-1] := 1.1e-18:
        V := U[N] - E:
```

```

if V<0 then
  ERROR(`E>U at right end:  increase R2.`);
end if;
V := V*EX[N] + DJ;
Y2 := P[N]*(1 - T*V);
V := (U[N-1] - E)*EX[N-1] + DJ;
Y3 := P[N-1]*(1 - T*V);
for i from 2 to N do
  j := N - i;
  Y1 := Y2;
  Y2 := Y3;
  Y3 := 2*Y2 - Y1 + W*V*P[j+1];
  V := (U[j] - E)*EX[j] + DJ;
  P[j] := Y3/(1 - T*V);
#   print(`J=`, j, `   Y=`, Y3, `   P=`, P[j]);
  if P[j] < P[j+1] then
    goto(80);
#   When psi begins to decrease, escape from loop and rescale psi
so that
#   outgoing psi matches ingoing psi[k]=1, with k as index for
which P[k]<P[k+1].
    end if;
  end do;
80: k := j;
  Q := 1/P[k];
  Y0 := Y2*Q;
  for i from k to N do
    P[i] := P[i]*Q;
  end do;
#   Start outward integration, from i=0 to i=k; then rescale psi so
that
#   outgoing psi matches ingoing psi at i=k; i.e. P[k] = 1.
  P[0] := 1.0e-18;
  P[1] := P[0]*evalf(exp(S*(J1 + 1/2)));
  V := (U[0] - E)*EX[0] + DJ;
  Y2 := P[0]*(1 - T*V);
  V := (U[1] - E)*EX[1] + DJ;
  Y3 := P[1]*(1 - T*V);
  for i from 2 to k do
    Y1 := Y2;
    Y2 := Y3;
    Y3 := 2*Y2 - Y1 + W*V*P[i-1];

```

```

    V := (U[i] - E)*EX[i] + DJ;
    P[i] := Y3/(1 - T*V);
#    print(`i=`, i, ` Y=`, Y3, ` P=`, P[i]);
end do;
Q := 1/P[k];
Y2 := Y2*Q;
Y3 := Y3*Q;
for i from 0 to k do
    P[i] := P[i]*Q;
end do;
G := 0;
for i from 0 to N do
    G := G + P[i]*P[i]*EX[i];
end do;
F := (2*Y3 - Y2 - Y0)/S^2 + C*V;
F := F/G/C;
E := E + F;
print(`dE = `, F);
# Calculate correction to E and return for another cycle;
#    repeat until dE < 0.3.
if abs(F) > 0.3 then
    goto(35);
end if;
# Count nodes to evaluate v.
NV := 0;
S1 := sign(P[0]);
for i from 1 to N do
    S2 := sign(P[i]);
    if S2<>S1 then
        NV := NV + 1;
        S1 := S2;
    end if;
end do;
Q := 1/sqrt(S*G);
# Normalise psi.
H := 0;
for i from 0 to N do
    Pe := P[i]*Q;
    TT := EX[i]*Pe;
    H := H + TT^2;
    P[i] := Pe;
end do;

```

```

H := H*S;
print(`v = `, NV, ` J = `, J1, ` E(v,J) = `, E);
print(` $\langle r^2 \rangle$  =`, H);
for i from 0 to N do
# print(`i = `, i, ` P[i] = `, P[i]);
end do;
PT := abs(P[N]);
if PT>0.00001 then
    print(" Warning -- |P[N]| > 1.0e-5; increase R2 and compare;
present P[N] =", P[N]);
end if;
PD := N/(NV + 1);
if PD<10 then
    print(" Warning -- less than 10 points on psi per node;
increase N and compare.");
end if;
# print(" values of r and Q(r) at both ends ");
# print(sqrt(EX[0]),P[0],sqrt(EX[1]),P[1],sqrt(EX[2]),P[2]);
# print(sqrt(EX[N]),P[N],sqrt(EX[N-1]),P[N-1],sqrt(EX[N-2]),P
[N-2]);
end proc:
> EigfunH(750,-10,4,0,-100000);
    dE =, -8735.2310712149370847671531088234
    dE =, -990.51736538154560915635705570244
    dE =, -11.562618808983039803167543358588
    dE =, -0.0017161337973984398081927478144675
    v =, 0, J =, 0, E(v,J) =, -109737.31277153926313216648590063
     $\langle r^2 \rangle$  =, 0.84008562314020992925417104097532
(13.28)

> EigfunH(750,-10,4,1,-20000);
    dE =, -4990.3923605264418838849869885196
    dE =, -2150.6045214836080456517725472988
    dE =, -288.39950544193364330810309625720
    dE =, -4.9309861576680013910089651986656
    dE =, -0.0016677487812209791838970345783209
    v =, 0, J =, 1, E(v,J) =, -27434.329041358432795215055494310
     $\langle r^2 \rangle$  =, 8.4008565661020931434565565221588
(13.29)

> EigfunH(750,-10,4,2,-10000);
    dE =, -1574.4610412038469348090480421645

```


$$\begin{aligned}
dE &=, -565.11458360840103309646279935955 \\
dE &=, -52.993761878366517307772154368787 \\
dE &=, -0.46568623167972765514450920696209 \\
dE &=, -0.000068133468320697749424326392130785 \\
v &=, 0, \quad J =, 2, \quad E(v,J) =, -12193.035141055762533566176929426 \\
\langle r^2 \rangle &=, 35.283593303078185424385797992622
\end{aligned}
\tag{13.30}$$

```

> with(ScientificConstants):
  Ri := GetValue(Constant(R[infinity]));
  evalf([-Ri/100, -Ri/400, -Ri/900], 12);
      Ri := 1.0973731568508 × 107
[ -109737.315685, -27434.3289212, -12193.0350761 ]

```

(13.31)

For H, the result for $J=0$ is nearly correct, but for $J=1$ or $J=2$ is exact, provided that the number of points and bounds of the range of integration are suitably chosen.

e3.72 exercise

Use the method of optimization with sequential simplices applied to data for the second virial coefficient of a specified pure monatomic gaseous substance to evaluate a function for potential energy; apply the two methods above for the numerical solution of Schrodinger's equation to calculate the energies of the vibrational states of the corresponding diatomic molecules, and estimate the errors of the differences between these vibrational energies for comparison with experimental data.

- i) Ar
- ii) other

3.8 introduction to quantum-chemical calculations

3.81 preparation for use of procedures

Within a context of fixed nuclear positions, quantum-chemical calculations have a useful function in chemistry, as long as one recognises the underlying assumptions and limitations. As a prelude to the operation of extensive programs for the calculation of molecular structure, we here present some procedures that illustrate the operation of those programs but with transparent coding; these procedures are modified from original work of Professor Hanspeter Huber and his students at University of Basel, Switzerland, and appear here with his permission.

The content of this section is adapted from material kindly provided by Professor Hanspeter Huber, including a paper of title *Transparent quantum chemistry* by P. S. Vogt, M. G. Müller and Hanspeter Huber of Institut für Physikalische Chemie der Universität Basel, Switzerland

transparent quantum chemistry

Transparent quantum chemistry

Patrick S. Vogt, Markus G. Mueller and Hanspeter Huber, Institut für Physikalische Chemie der Universität Basel, Klingelbergstrasse 80, CH-4056 Basel, Switzerland

abstract

To teach the techniques of quantum-chemical methods, *Maple* procedures are presented in three suites. These suites comprise related procedures to calculate electronic properties of one-electron systems, further calculations at levels SCF, MP2 and CI, and with density-functional theory. These suites might serve both instructors for demonstrations and students who thereby become familiar with the techniques on observing a calculation each step at a time. The properties of *Maple* allow both an overview of the structure of the programs and detailed scrutiny of the techniques employed in such calculations. The visibility of each step of a calculation and of the parameter values, the associated comments, the possibility to document a calculation or to implement changes for any desired purpose make these suites a useful tool. The presented examples require duration of execution between a few seconds and a few minutes on a typical computer. This property, together with the widespread availability of *Maple*, enables students directly to perform exercises in quantum chemistry.

key words quantum chemistry, computational chemistry, theoretical chemistry, computer-assisted instruction, teaching and learning aids

introduction

Before year 1970, quantum-chemical calculations were reserved for a few specialists. With the advent of calculations so called semi-empirical, such as those known by their acronyms CNDO, INDO et cetera, and major packages such as *Gaussian*, the situation altered a few years thereafter. A wider audience began to apply quantum-chemical methods and became familiar with the underlying theory. The main users of the programs remained, however, physical and theoretical chemists, who had some familiarity with both the use of computers and the handling of the numeric input. This condition became altered substantially after 1985, at which time graphic interfaces made a common computer a ubiquitous personal tool, and programs such as *Spartan* enabled quantum-chemical calculations to be undertaken by undergraduate students. This development is useful providing that it is accompanied by courses or books (e.g. ref. 1), which teach how the programs should be applied and which methods and basis sets are adequate for a particular purpose. The disadvantage of these developments is that the number of persons who understand the underlying theory has diminished to a few specialists.

The programs presented here are intended for courses that provide a background to the quantum-chemical methods, for which the 'black-box' program packages are initially unsuitable. Instructors might use the present suites of procedures for demonstrations and students for exercises to observe each step of a calculation. To make the programs simple and readily understandable, all intricacies, acceleration methods and so forth are avoided. As far as practicable, similar structures are used in separate suites to facilitate the discerning by students of the similarities between the methods. The suites for electronic calculations employ only spherical gaussian functions and without contractions; these are centred independently from the nuclei, which for instance allows the inclusion of polarization or calculations with occupied p orbitals. The modified set contains these three suites,

- one-electron system,
- self-consistent field, second-order Moller-Plesset perturbation theory and configuration interaction, and
- density-functional theory,

of which the latter suite is restricted to atoms. The calculations for Moller-Plesset perturbation theory of second order (MP2) and for configuration interaction, originally in separate suites, are here combined with those for the self-consistent field: the output from one stage serves as input to the next stage within the same suite. The use of density functionals requires an integration over the functional in three dimensions, which might execute slowly for a non-spherical system; for a rigorously spherical system, such a calculation becomes essentially an integration over the radius in one dimension, thus simplifying programming and speeding execution. Although density functionals are applied typically in calculations on large molecules and are known to yield poor results for small molecules in some cases, this method

remains useful for illustrative purposes, and might serve for comparison with results from a calculation on He for a self-consistent field, Moller-Plesset perturbation theory and configuration interaction; as the density-functional theory is supposed to take into account, at least partially, the correlation between electrons, similar to the effect of perturbation theory at the level MP2, the results of a calculation based on density-functional theory might be directly compared with those from perturbation theory applied after the attainment of a self-consistent field. With procedures in a separate suite, the one-dimensional Schrödinger equation is solvable for any function of potential energy according to the renormalized Numerov method [2,3]; this capability might be applied, for instance, to calculate a vibrational spectrum of a diatomic molecule from a function of potential energy and the electric dipolar moment that had been previously calculated pointwise with the MP2 procedure and fitted to an algebraic form. As the *Maple* procedures are executable serialtim, all variables are accessible and each step is visible. Detailed comments accompany the procedures, but the underlying theory is presumed. In what follows we present a few examples of applications of the suites of procedures.

matrix formalism

The algebraic solution of Schrödinger's temporally independent equation for the hydrogen atom is presented in many books and introductory lectures on quantum chemistry. Although this point is useful for initiation, it is well recognized to be inappropriate with respect to any other atomic or molecular calculation. A superior approach begins with a gaussian function, $\psi(r)$, with exponent α in three-dimensional space as a basis function,

$$\psi(r) = N_o e^{-\alpha r^2}$$

and to calculate its energy expectation value, E ,

$$E = \langle \psi(r) | H | \psi(r) \rangle = \langle \psi(r) | T | \psi(r) \rangle + \langle \psi(r) | V | \psi(r) \rangle = \frac{3\alpha}{2} - 2\sqrt{\frac{2\alpha}{\pi}}$$

in which appear normalizing factor N_o and operators for the hamiltonian H , kinetic energy T and potential energy V ; the integrals of the kinetic-energy and potential-energy operators for the general case might be given at this point in a lecture, as they become applied subsequently. Setting the derivative of

E with respect to α equal to zero, so $\frac{\partial}{\partial \alpha} E = 0$, yields an optimal energy

$$E = -\frac{4}{3\pi} E_h \sim -0.4244 E_h$$

for this basis function, to be compared with the exact value $-0.5 E_h$; here E_h denotes a unit of energy, called hartree, according to the system of atomic units: $E_h = 4.35974417 \cdot 10^{-18}$ J. In the next step, two gaussian basis functions in a linear combination might be employed, showing the difficulties that might occur if one seeks to solve for optimal coefficients and exponents. We hence apply a 'computational' approach: a Monte-Carlo optimization with the user as a random generator, which is implemented in a Java-Script-based HTML document available through internet at the address

<http://www.chemie.unibas.ch/PC/Huber/HAtom/HAtom.htm>

The student will soon discover that this approach leads nowhere, which is an effective argument to switch to a matrix approach in which only the linear coefficients are optimized. With the *Maple* procedures in the suite `one_elec`, the exercise is then repeated with two basis functions; the same result as above is obtained; note that the *Maple* suite uses normalized gaussian functions in contrast to the above JavaScript program that uses unnormalized functions -- this condition must be considered in a comparison of the coefficients. As shown in Fig. 2, *Maple* permits the user to show only the program structure, i.e. the input section, definition of parameters and auxiliary functions, integral functions, integral or operator matrices and solution of the eigenvalue problem, respectively. The user can

subsequently investigate each in as much detail as desired.

Larger basis sets, e.g. from Huzinaga [4] or Schmidt and Ruedenberg [5], might be subsequently used to show convergence to the exact solution. Another effective exercise is to compare the results of two basis sets of equal size, an optimized basis by Huzinaga (exponents 68.16, 10.2465, 2.34648, 0.67332, 0.22466 and 0.082217) and an even-tempered basis with more diffuse functions (exponents 10.42, 2.56, 0.64, 0.16, 0.04 and 0.01). One finds that the former yields an improved value for the lowest orbital, but poor values for the higher orbitals, and vice versa. These results provide an opportunity to discuss the qualities of basis sets and their relation to a desired property. H_3^{2+} , arranged linearly and in an equilateral triangle, provides insight into the dependence of the electronic structure on the molecular geometry and might serve for a comparison with the well known rule for the purported π system of orbitals of cyclic molecules such as benzene.

the role of polarization functions

To show the role of polarization functions, we use the simple system H_2^{+} , having one electron, at its experimental equilibrium internuclear distance $1.988 a_o$. We apply first even-tempered basis sets [5] of increasing size on both hydrogen nuclei until we reach convergence. The electronic energy converges towards $-1.094 E_h$ and the total energy towards $-0.591 E_h$, in contrast to the experimental value $-0.603 E_h$. Students of chemistry are typically aware that electron 'clouds' are polarized by neighbouring nuclei; they are thus readily convinced that providing spherical basis functions located on only the hydrogens fails to allow the electron clouds to polarize towards the centre of the bond, but by shifting some functions from the nuclei towards the centre of a bond we allow for polarization. This strategy is implemented in an exercise with the following basis set:

$\frac{\alpha}{a_o^2}$	293.57	43.93	9.8912	2.7652	1.9	0.79908	0.33607	0.14134	0.05944	0.025
$\frac{\delta x}{a_o}$	0	0	0	0	0.347	0.1434	0.25	0.295	0.425	0.5

in which α is the exponent and δx is the shift of the basis function away from the nucleus toward the centre of the bond; in general, this basis set must be applied twice, i.e. to each atom. The input section for a calculation including polarization functions shifted along the bond follows:

- **Digits** the accuracy; in some cases an increase might be useful
- **NumberOfNuclei** the total number of atomic centres
- **Dim** the order of the principal matrices, i.e. the sum of all basis functions, in this case 20 as we have 10 for each nucleus
- **AtomicNumber** an array containing the atomic numbers of all atoms
- **NuclearCoordinate** a two-dimensional array containing coordinate set x, y, z of all nuclei in the same order as defined in the array of atomic numbers
- **Exponent** an array containing the exponents of all basis functions; as two equal sets of ten functions are used, the array contains 20 exponents as ten pairs of identical exponents
- **BasisCoordinate** a two-dimensional array containing coordinate sets x, y, z of all basis functions in the same order as defined in the array of the exponents. As we have defined first the ten functions corresponding to the nucleus in the origin of the coordinate system, we must specify first the corresponding coordinate sets. For example, the first four basis functions are located on the first

nucleus and the next are then shifted by the amounts given in the exercise, et cetera; the output matrix is not shown in the figure.

> # figure 1

The resulting energy is now $-0.603 E_h$ (electronic $-1.106 E_h$) in satisfactory agreement with experiment.

This example again demonstrates the influence of the quality of basis sets. The present basis set provides a virtually exact calculated potential curve, as is shown below. To float spherical functions away from the nuclei is not a standard strategy to introduce polarization, but one can readily appreciate that p functions along the bond axis permit the formation of sp hybrids, which have the same effect. At this point one might note that the inclusion of a p basis function has nothing to do with an occupied p molecular orbital.

inclusion of atoms with occupied p orbitals

In programs utilizing only lobe functions [6,7], a p_x function is constructed on shifting two spherical gaussians away from the centre, one by positive δx and another by negative δx . These functions are contracted with coefficients of the same size but different sign. In the present suites contractions are not implemented, but this approach is possible. In orbitals of type p, the corresponding coefficients are of equal size but different sign, as a result of the calculation. For molecular orbitals of other symmetries they contribute differently.

As an example, an SCF calculation for water at the experimental geometry yields an energy $-75.0359 E_h$ after six iterations, utilizing one 1s basis function (exponent 1.0) on each hydrogen, one 1s (exponent 20.0), one 2s (exponent 1.0) and three p functions, implemented as six spherical basis functions (exponent 0.7, two of them placed on each axis with shifts $\pm 0.1 a_o$), on oxygen. To show the symmetry in the coefficients better, it is recommended to increase the number of digits with `Digits:=20` in the input section.

calculated anharmonic wavenumbers for vibrations of a diatomic molecule

Whereas harmonic wavenumbers are available in calculations with packages such as *Gaussian*, generally calculated as algebraic second derivatives, anharmonic wavenumbers are typically inaccessible. *Maple* procedures in the present suites facilitate the calculation of a curve for potential energy, e.g. of H_2 or H_2^+ , point-wise with reasonable accuracy. For a class of ten students for example, each student might perform an SCF or MP2 calculation at a given internuclear distance, hence yielding ten points of the potential-energy curve. An algebraic curve is then fitted to these points, and implemented in procedures of the suite *Schrödl*. Figure 4 shows the program section for a potential of H_2^+ .

At the beginning the parameters of the potential-energy function in algebraic form are defined. The central section is used to define parameters that are needed for plotting. The potential occurs in the loop at the end, at which it serves to fill the array `Acquisition[Potential,k]` with k equal to `max_n_dim` points of the potential function; the first line in the loop shifts and scales the x -axis such that a reasonable part of the potential is used and plotted. The result of such a calculation with the

potential energy in figure 4 is shown in figure 5. The wavenumbers obtained from $v = \frac{\Delta E}{h}$ are 2219,

2066, 1932 and 1809 cm^{-1} , respectively (experimental: 2189, 2057, 1925 and 1792 cm^{-1}). These values demonstrate clearly the anharmonicity of such a potential-energy function. Taking the first and second derivatives of the algebraic curve and a few simple manipulations yield the equilibrium distance, the dissociation energy and the harmonic wavenumbers for comparison with experiment.

insight into the self-consistent field and beyond

As all details of a calculation are visible with procedures in these suites, it is practicable to test some properties that are generally inaccessible. For example, in a simple calculation of a self-consistent field (restricted Hartree-Fock), matrix elements H_{ii} for a one-electron hamiltonian might be compared with Coulomb integrals J_{ii} , which in calculations of Hückel type are neglected. Such a study is also instructive for a dissociation curve of H_2 at the SCF level, giving insight into the breakdown of this method at large distances.

Whereas the MP2 procedure automatically calculates the energies of Moller-Plesset perturbation theory of second order based on the input from a converged SCF sheet, the procedure for configuration interaction requires further input from a user in the last section "Construct CIMatrix". The user must decide which configurations should be included and must find the necessary matrix elements, such as $2 H_{ii} + J_{ii}$, (12/12), (23/23), $H_{22} + H_{33} + J_{23}$ or $H_{23} + (33/23)$. The one-electron integrals are called as matrix elements, e.g. $\langle \psi_2 | H_{core} | \psi_3 \rangle = H_{23}$ is $H_{ij} [2, 3]$, whereas the two-electron integrals are function calls; e.g. $\langle \psi_2 \psi_3 | \frac{1}{r} | \psi_2 \psi_3 \rangle$ (in Mulliken's nomenclature (22/33)) is $ijklIntegral(2, 2, 3, 3)$. The student must hence think before initiating the calculation. To obtain a satisfactory correlation energy, it is recommended to use p functions emulated in the same way as the polarization functions but with larger exponents.

conclusion

With these *Maple* suites we contribute to the teaching of the foundations of quantum-chemical methods. The procedures are lean and execute the computational work for the student, but he or she must still do some thinking. The properties of *Maple* allow both an overview of the structure of the suites of procedures and a detailed study of technical questions, concerning how such calculations are undertaken. The visibility of each step of a calculation and of the values of all parameters, the associated comments, the possibility for documentation of a calculation and to implement changes for a desired purpose should make them a useful tool. The presented examples take durations between seconds and a few minutes on a typical ubiquitous computer. Together with the widespread availability of *Maple*, this property enables performing exercises even at home.

references

- 1 W. J. Hehre, L. Radom, P. von R. Schleyer, J. A. Pople *Ab initio Molecular Orbital Theory*, New York, USA: Wiley, 1986
- 2 J. M. Blatt *J. Comput. Phys.* 1, 382 -- 396, 1967
- 3 B. R. Johnson *J. Chem. Phys.* 67, 4086 -- 4093, 1977
- 4 S. Huzinaga *J. Chem. Phys.* 42, 1293 -- 1302, 1965
- 5 M. W. Schmidt, K. Ruedenberg *J. Chem. Phys.* 71, 3951 -- 3962, 1979
- 6 H. Preuss *Z. Naturforsch.* a11, 823 -- 831, 1956
- 7 J. L. Whitten *J. Chem. Phys.* 44, 359 -- 364, 1966

remarks concerning configuration interaction (CI)

Annex I -- Matrix Elements for a CI Calculation -- and annex II -- Example of a CI calculation: He with five basis functions -- are appended below. The first equation in I shows that we form a linear combination of configurations: the number of configurations that we include is arbitrary and bears no relation to the number of basis functions used; in II these numbers are fortuitously identical. For He in II we used five basis functions, which results in five molecular orbitals, but the first graph shows that we use only the lowest three to form configurations. If we included all possible excitations, i.e. configurations with all five molecular orbitals, we would call it a 'full CI' or 'FCI'. For He I we use no

singly excited configuration because its contribution is small. It can be shown that, if the molecular orbitals are from an SCF calculation, a matrix element between the ground and any singly excited state is zero; for this reason the singly excited configurations contribute only through indirect coupling with other matrix elements, which is an effect of second order and thus much smaller. We use this example as a test for understanding.

$$E(\text{SCF}) = -2.8586 E_h, \quad E(\text{CI}) = -2.8752 E_h, \quad E(\text{MP2}) = -2.8719 E_h$$

In the upper part of I the general CI matrix must be diagonalized exactly as in Hückel molecular-orbital theory, but in contrast to that Hückel theory the matrix elements are formed not with orbitals but with configurations; how this is done is illustrated below. In the lower part of II the matrix elements correspond to the configurations for helium.

For HLi one might take only the three lowest orbitals, keep the lowest two electrons in the lowest orbital and make all possible excitations of the upper two, in which α and β denote spins, or values of quantum number $m_s = \frac{1}{2}$ or $-\frac{1}{2}$ respectively with respect to some axis:

$$\begin{array}{cccc} \text{---} & \text{---} \alpha \beta \text{---} & \text{---} \beta \text{---} & \text{---} \beta \text{---} \\ \text{---} \alpha \beta \text{---} & \text{---} & \text{---} \alpha \text{---} & \text{---} \alpha \text{---} \\ \text{---} \alpha \beta \text{---} & \text{---} \alpha \beta \text{---} & \text{---} \alpha \beta \text{---} & \text{---} \alpha \beta \text{---} \\ \psi_0 & \psi_1 & \psi_2 & \psi_3 \end{array}$$

(This choice is likely not optimal, as the last two configurations are singly excited and therefore contribute little, but this case is merely for explanatory purposes). The corresponding matrix elements are found in the suite of *Maple* procedures. Note that $H_{ab} \rightarrow \text{Hi j}[a, b]$; $J_{ab} = (aa|bb) \rightarrow$

`IJKLIntegral(a, a, b, b)`; $K_{ab} = (ab|ab) \rightarrow \text{IJKLIntegral}(a, b, a, b); $(ab|cd) \rightarrow$$

`IJKLIntegral(a, b, c, d)`; (...|...) is the Mulliken notation.

Here follows a detailed explanation how to find the matrix elements. The equations in I are for spin-orbitals, i.e. the product of the space orbital and the spin. We call the space orbitals from bottom to top 1, 2 and 3, and the corresponding spin-orbitals

$$a = \psi_1 \alpha, \quad b = \psi_1 \beta, \quad c = \psi_2 \alpha, \quad d = \psi_2 \beta, \quad e = \psi_3 \alpha, \quad f = \psi_3 \beta.$$

example 1, $\langle \psi_1 | H | \psi_1 \rangle$: as the second configuration is the same as the first, it constitutes the 'non-excited' case.

a) The first sum yields $H_{aa} + H_{bb} + H_{ee} + H_{ff}$, with a term for each occupied spin-orbital.

b) The second term yields $\frac{1}{2} (J_{ab} + J_{ae} + J_{af} + J_{ba} + J_{be} + J_{bf} + J_{ea} + J_{eb} + J_{ef} + J_{fa} + J_{fb} + J_{fe})$

c) The third term yields $-\frac{1}{2} (K_{ae} + K_{ca})$ and the fourth $-\frac{1}{2} (K_{bf} + K_{fb})$; the sums here are over occupied orbitals of the same spin only).

On renaming to space orbitals, a) becomes $2 H_{11} + 2 H_{33}$, b) becomes $J_{11} + J_{33} + 4 J_{13}$, c) becomes $-2 K_{13}$; these terms are what are programmed for the matrix element $\langle \psi_1 | H | \psi_1 \rangle$.

example 2, $\langle \psi_0 | H | \psi_1 \rangle$: as the second configuration is doubly excited relative to the first, it is the case 'doubly excited'. $m \rightarrow p, k \rightarrow l$ as in our case $c \rightarrow e, d \rightarrow f$, which yields $(ce|df) - (cf|ed)$. The second integral is zero because of spin integration, as c and f , like e and d , have not the same spin; note that $\langle \alpha|\alpha \rangle = \langle \beta|\beta \rangle = 1, \langle \alpha|\beta \rangle = \langle \beta|\alpha \rangle = 0$. The remaining integral is renamed to (23|23) using the nomenclature for space orbitals. Note that $(ab|cd) = (ba|cd) = (ba|dc) = (ab|dc) = (cd|ab)$ et cetera.

example 3, $\langle \psi_0 | H | \psi_2 \rangle$: as the second configuration is singly excited relative to the first, it is the case

'singly excited'.

$m \rightarrow p$ is in this case $d \rightarrow f$, which yields $H_{df} + (aa|df) + (bb|df) + (cc|df) - (bd|bf)$; note that $(ad|af) = (cd|cf) = 0$ through spin integration). The renaming yields $H_{23} + 2(11|23) + 2(22|23) - (12|13)$

Together with the *Maple* procedures, these examples assist an understanding of the construction of the matrix. On examining the results from the *Maple* calculation, one sees that elements 1,3 and 1,4 are virtually zero as expected for the singly excited configurations. The CI energy appears as the lowest eigenvalue - 8.92043 E_h . The electric dipolar moment may be taken as the value from the SCF calculation; to calculate the CI moment is much complicated. The quantity $\langle \Psi_0 | H | \Psi_0 \rangle$ is exactly the SCF energy.

annex I Matrix elements to calculate configuration interaction

Here we treat spin-orbitals, for which the indices run over spin-orbitals,

$$\text{ansatz} \quad \Psi_j = \sum_i c_{i,j} \psi_i$$

in which amplitude function Ψ_j for state j is taken as a sum of configurations ψ_i , which is a Slater determinant of molecular orbitals ϕ_k and which is weighted according the coefficients $c_{i,j}$. Variation yields the following matrix; note that the configurations are kept constant, i.e. everything is formally analogous to the one-electron-system:

$$\begin{bmatrix} \Psi_0 | H | \Psi_0 - E & \Psi_0 | H | \Psi_1 & \Psi_0 | H | \Psi_2 & \dots \\ \Psi_1 | H | \Psi_0 & \Psi_1 | H | \Psi_1 - E & \Psi_1 | H | \Psi_2 & \dots \\ \Psi_2 | H | \Psi_0 & \Psi_2 | H | \Psi_1 & \Psi_2 | H | \Psi_2 - E & \dots \\ \dots & \dots & \dots & \dots \end{bmatrix}$$

The integrals $\langle \psi_i | H | \psi_j \rangle$ in each matrix element are here no longer over molecular orbitals, but over configurations. Because configurations correspond to Slater determinants, the derivation is complicated, but the simplest, that for the ground configuration $\langle \Psi_0 | H | \Psi_0 \rangle$, is already derived to obtain the ground state energy at the stage of self-consistent field.

1. non-excited

$$\langle \Psi_0 | H | \Psi_0 \rangle = \sum_{i=1}^{occ} H_{ii} + \frac{1}{2} \sum_{i=j}^{occ} \sum_{i=j}^{occ} J_{ij} - \frac{1}{2} \sum_{i=j}^{\alpha} \sum_{i=j}^{\alpha} K_{ij} - \frac{1}{2} \sum_{i=j}^{\beta} \sum_{i=j}^{\beta} K_{ij}$$

in which $J_{ij} = (ii|jj)$ and $K_{ij} = (ij|ij)$ and " $j = <>i$ " signifies that the case $j=i$ is excluded from the sum.

2. singly excited

$$\langle \Psi_0 | H | \Psi_1 \rangle = H_{mp} + \sum_{i=1}^{occ} (ii|mp) - (ip|im) \quad (\text{integrate spins})$$

in which " $i = <>m, p$ " signifies that the cases $i=m$ and $i=p$ are excluded from the sum.

3. doubly excited

$$\langle \Psi_0 | H | \Psi_2 \rangle = (mp|kl) - (ml|pk) \quad (\text{integrate spins})$$

4. more highly excited

$$\langle \psi_0 | H | \psi_3 \rangle = 0$$

annex II Example of a CI calculation: He with five basis functions

The exponents of the primitive basis functions are: 72.5856, 18.1174, 4.52212, 1.12872 and 0.28173. An SCF calculation is performed and yields an energy $-2.8586 E_h$ for the ground configuration. Which configurations should we choose for the CI calculation?

Here is an arbitrary choice,

					MO
_____	_____	_____	_____	_____	5
_____	_____	_____	_____	_____	4
_____	_____	<u>αβ</u>	<u>α</u>	<u>β</u>	3
_____	<u>αβ</u>	_____	<u>β</u>	<u>α</u>	2
<u>αβ</u>	_____	_____	_____	_____	1
ψ_0	ψ_1	ψ_2	ψ_3	ψ_4	

in which α and β signify spins. Excitations to the molecular orbitals 4 and 5 are neglected, because the greater is the energy, the smaller is the contribution. According to a theorem, singly excited configurations, beginning from an SCF configuration, form a zero matrix element with the ground state; singly excited configurations hence yield no direct contribution to energy, but they can contribute to a smaller extent through an indirect interaction, e.g. with doubly excited configurations. Triplets are neglected because they do not interact with singlets.

This choice yields the following CI matrix:

	ψ_0	ψ_1	ψ_2	ψ_3	ψ_4
ψ_0	$2 H_{11} + J_{11}$	$(21 21)$	(3131)	$(31 21)$	$(31 21)$
ψ_1		$2 H_{22} + J_{22}$	$(32 32)$	$H_{23} + (32 22)$	$H_{23} + (32 22)$
ψ_2			$2 H_{33} + J_{33}$	$H_{23} + (33 32)$	$H_{23} + (33 32)$
ψ_3				$H_{22} + H_{33} + J_{23}$	$(32 32)$
ψ_4					$H_{22} + H_{33} + J_{23}$

The column at the left is not a matrix, merely a column of symbols to indicate the relation to the elements of this square matrix of order 5. The matrix element for a singly excited configuration -- an excitation from the ground state into the first MO -- would be $H_{12} + (21|11)$. See later for its size in our example, according to the above theorem.

From Hanspeter Huber <Hanspeter.Huber@unibas.ch>

Subject: *Maple* procedures for quantum chemistry

Q In relation to the calculation of electronic energies, I had an idea that, for pedagogical purposes, it might be interesting to use the output from the electronic calculations for diatomic molecules as input, in suitably fitted form, for the vibrational calculation.

A It is a nice idea. I do not know whether *Maple* includes a suitable fitting procedure?

Q When I examined the electronic procedures, I was unable to find evidence for the calculation of the internuclear coulombic repulsion term, which is of course needed to combine with the electronic energy to form the total energy in the electronic problem that becomes the potential energy for the vibrational problem.

A You are right. I did not include it in the sheet, but point out to the students that this is only the electronic energy and that they must add the nuclear energy for the total energy as in a potential energy, which they can easily calculate with a pocket calculator. Of course what happens in nearly all exercises is that they forget it and are astonished about the strange behaviour of the curve. I do this because I think that they better remember subsequently that the nuclear part is a simple coulombic term (so learning

from errors). For your application above, it is reasonable to include it.

Q When, for H_2^+ in the worksheet OneElec.mws, I tried to add the smallest orbital energy to the nuclear repulsion term, just $\frac{1}{R}$ in atomic units, I failed to obtain anything that seemed to resemble the points on a customary curve for potential energy. Where is the error?

A I see nothing wrong in what you did -- it is what I would also do. Perhaps you made just some technical error. I calculated a few points (with your and my sheet, which gave the same results) and present here the results; perhaps it helps you to find the error:

Distance $/a_0$	electronic energy	nuclear energy ($\frac{1}{R}$)	total energy (sum)	total energy relative to dissociation
0.5	-1.653446	2.0	0.346554	0.832366
1.5	-1.204639	0.666666	-0.537972	-0.052160
5.0	-0.706278	0.2	-0.506278	-0.020466
20.0	-0.538492	0.05	-0.488492	-0.002680

It is the second example on your sheet with the small basis set (1.335|0.2018). The last column is relative to dissociation into H^+ and H-atom, i.e. I have added 0.485812, which is the energy obtained for a H-atom with the same basis set.

(By the way, could it be that you used `Dim:=2` instead of `Dim:=4`? I just found that our comment there is not so clear as `Dim` is the total number of basis sets on all atoms. I mention this, because you give the second example with `#`, but for `Dim` you do not give `#4`)

3.82 one-electron program for small molecules

program for small molecules with one electron, utilizing basis sets comprising spherical gaussians, developed and copyright: P. Vogt, H. Huber, Hanspeter.Huber, 1999 November. All properties are expressed in atomic units!

input (nuclear data, basis-set data)

```
> restart:
  with(linalg):
> Digits := 10:          # numerical accuracy
```

For this example, we test the hydrogen atom, H, so one atom or atomic nucleus, with two basis functions located on that nucleus.

```
> NumberofNuclei := 1:
  Dim := 2:
```

We define the nuclei with its atomic number, and specify the location of the nucleus at the origin of the coordinate system.

```
> AtomicNumber := array([1]);
  NuclearCoordinate := array([[0,0,0]]);
  AtomicNumber := [ 1 ]
```

$$\text{NuclearCoordinate} := \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \quad (16.1.1)$$

To define basis set, we provide the values of α in the exponents and the positions of the basis functions as cartesian coordinates.

```
> Exponent := array([1.335,0.2018]);
```

$$\text{Exponent} := \begin{bmatrix} 1.335 & 0.2018 \end{bmatrix} \quad (16.1.2)$$

```
> BasisCoordinate := array([[0,0,0], [0,0,0]]);
```

$$\text{BasisCoordinate} := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (16.1.3)$$

parameters and auxiliary functions

To make the program more readable, coordinates x , y and z are stored in arrays.

```
> X := 1:
  Y := 2:
  Z := 3:
```

sqrDistance - Function to calculate the square of the distance between two basis functions

```
> sqrDistance := proc(func1, func2)
  (BasisCoordinate[func1,X]-BasisCoordinate[func2,X])^2 +
  (BasisCoordinate[func1,Y]-BasisCoordinate[func2,Y])^2 +
  (BasisCoordinate[func1,Z]-BasisCoordinate[func2,Z])^2;
end proc;
```

ProductGaussian - Calculates the product of two Gaussian functions

The product of two Gaussian functions is again a Gaussian with a new exponent c_{new} , which is the sum of the original exponents, and new coordinates $R_{x\text{new}}$, $R_{y\text{new}}$, $R_{z\text{new}}$, which are found from the old ones by weighting with the exponents.

```
> ProductGaussian := proc(func1,func2,cnew::evaln,Rxnew::evaln,
  Rynew::evaln,Rznew::evaln)
  local a,b;
  cnew := Exponent[func1] + Exponent[func2];
  a := Exponent[func1] / eval(cnew);
  b := Exponent[func2] / eval(cnew);
  Rxnew := a * BasisCoordinate[func1,X] + b * BasisCoordinate
[func2,X];
  Rynew := a * BasisCoordinate[func1,Y] + b * BasisCoordinate
[func2,Y];
  Rznew := a * BasisCoordinate[func1,Z] + b * BasisCoordinate
[func2,Z];
end proc;
```

AuxInt - Auxiliary function for the calculation of the potential integrals

```
> AuxInt := proc(X)
  if X=0 then
```

```

1
else
    evalf(1/2*sqrt(Pi/X)*erf(sqrt(X)))
end if;
end proc:

```

diagonalisation of a matrix by the Jacobi method

a: matrix to be diagonalised

d: eigenvalues

v: eigenvectors

includes symmetrisation of the matrix at the beginning and sorting of the eigenvalues and eigenvectors at the end

```

> Diagonalisation := proc(a,d,v)
    local i,ip,iq,j,c,g,h,s,sm,t,tau,theta,tresh,b,z,Maxiter;
    Symmetrisation(a);
    Maxiter := 10^Digits;
    d := array(1..Dim);
    v := array(1..Dim,1..Dim);
    b := array(1..Dim);
    z := array(1..Dim);
    for ip from 1 to Dim do
        for iq from 1 to Dim do
            v[ip,iq] := 0;
        end do:
        v[ip,ip] := 1;
    end do:
    for ip from 1 to Dim do
        b[ip] := a[ip,ip];
        d[ip] := b[ip];
        z[ip] := 0;
    end do:
    for i from 1 to Maxiter do
        sm := 0;
        for ip from 1 to Dim-1 do
            for iq from ip+1 to Dim do
                sm := sm + abs(a[ip,iq]);
            end do:
        end do:
        if(sm<10^(-Digits-2)) then
            break
        end if;
        if i<4 then
            tresh := 0.2*sm/Dim^2;
        else

```

```

    tresh := 0;
end if;
for ip from 1 to Dim-1 do
    for iq from ip+1 to Dim do
        g := 100*abs(a[ip,iq]);
        if i>4 and abs(d[ip])+g = abs(d[ip]) and abs(d[iq])+g =
abs(d[iq]) then
            a[ip,iq] := 0;
        elif abs(a[ip,iq]) > tresh then
            h := evalm(d[iq] - d[ip]);
            if abs(h)+g = abs(h) then
                t := a[ip,iq]/h;
            else
                theta := 0.5*h/a[ip,iq];
                t := 1/(abs(theta)+sqrt(1.+theta^2));
                if theta < 0 then
                    t := -t;
                end if;
            end if;
            c := 1/sqrt(1+t^2);
            s := t*c;
            tau := s/(1+c);
            h := t*a[ip,iq];
            z[ip] := z[ip] - h;
            z[iq] := z[iq] + h;
            d[ip] := d[ip] - h;
            d[iq] := d[iq] + h;
            a[ip,iq] := 0;
            for j from 1 to ip-1 do
                g := a[j,ip];
                h := a[j,iq];
                a[j,ip] := g - s*(h+g*tau);
                a[j,iq] := h + s*(g-h*tau);
            end do;
            for j from ip+1 to iq-1 do
                g := a[ip,j];
                h := a[j,iq];
                a[ip,j] := g - s*(h + g*tau);
                a[j,iq] := h + s*(g - h*tau);
            end do;
            for j from iq+1 to Dim do
                g := a[ip,j]:

```

```

        h := a[iq,j]:
        a[ip,j] := g - s*(h + g*tau):
        a[iq,j] := h + s*(g - h*tau):
    end do:
    for j from 1 to Dim do
        g := v[j,ip]:
        h := v[j,iq]:
        v[j,ip] := g - s*(h + g*tau):
        v[j,iq] := h + s*(g - h*tau):
    end do:
    end if:
end do:
for ip from 1 to Dim do
    b[ip] := b[ip] + z[ip]:
    d[ip] := b[ip]:
    z[ip] := 0:
end do:
end do:
Sort(d,v);
print(d);
end proc:

```

sort the eigenvalues und eigenvectors

To sort the eigenvalues and corresponding eigenvectors after a diagonalisation according to the size of the eigenvalues.

```

> Sort := proc(Eigenvalues,Eigenvect)
    local MaxValue,MaxIndex,column1,column2, buffer;
    for column1 to Dim-1 do
        column1;
        MaxValue := Eigenvalues[column1];
        MaxIndex := Dim + 1;
        for column2 from column1+1 to Dim do
            if MaxValue > Eigenvalues[column2] then
                MaxIndex := column2;
                MaxValue := Eigenvalues[MaxIndex];
            end if;
        end do;
        if MaxIndex < Dim+1 then
            Eigenvect := swapcol(Eigenvect,column1,MaxIndex);
            Eigenvalues[MaxIndex] := Eigenvalues[column1];
            Eigenvalues[column1] := MaxValue;
        end if;
    end do;
end proc:

```

```

    end do;
    evalm(Eigenvalues);
end proc:

```

Symmetrisation - Symmetrizes a matrix A

```

> Symmetrisation := proc(A)
    A := scalarmul(matadd(A,transpose(A)),0.5);
end proc:

```

integral functions (S, T, V)

Overlapintegral - function to calculate the overlapintegrals $\langle \phi | \phi \rangle$

```

> Overlapintegral := proc (func1, func2)
    local alpha, beta, cinv, Q, aux;
    alpha := Exponent[func1];
    beta := Exponent[func2];
    cinv := 1/(alpha+beta);
    Q := exp(-alpha * beta * cinv * sqrDistance(func1,func2));
    aux := (4*alpha*beta*cin^2);
    Q*sqrt(sqrt(aux^3));
end proc:

```

KineticIntegral - function to calculate the kinetic integral $\langle \phi | T | \phi \rangle$

```

> KineticIntegral := proc (func1, func2)
    local alpha, beta, E;
    alpha := Exponent[func1];
    beta := Exponent[func2];
    E := alpha*beta/(alpha + beta);
    Overlapintegral(func1,func2)*E*(3 - 2*E*sqrDistance(func1,func2)
);
end proc:

```

PotentialIntegral - function to calculate the potential integral $\langle \phi | V | \phi \rangle$

```

> PotentialIntegral := proc(func1, func2)
    local c, Rx, Ry, Rz, V, argument, nuc;
    ProductGaussian(func1,func2,c,Rx,Ry,Rz);
    V := 0;
    for nuc to NumberofNuclei do
        argument := c*((Rx-NuclearCoordinate[nuc,X])^2 +
            (Ry-NuclearCoordinate[nuc,Y])^2 + (Rz-NuclearCoordinate
[nuc,Z])^2);
        V := V + AtomicNumber[nuc] * AuxInt(argument);
    end do;
    evalf(-Overlapintegral(func1,func2) * 2/sqrt(Pi) * sqrt(c) * V);

```

end proc:

construction of integral matrices

overlap matrix S

```
> SMatrix := array(1..Dim,1..Dim):  
  for lrow to Dim do  
    for lcolumn to lrow do  
      SMatrix[lrow,lcolumn] := OverlapIntegral(lrow,lcolumn);  
      SMatrix[lcolumn,lrow] := SMatrix[lrow,lcolumn];  
    end do;  
  end do;  
  print (SMatrix):
```

$$\begin{bmatrix} 1.000000000 & 0.5551633485 \\ 0.5551633485 & 1.000000000 \end{bmatrix}$$

(16.4.1.1)

kinetic energy matrix T

```
> TMatrix := array(1..Dim,1..Dim):  
  for lrow to Dim do  
    for lcolumn to lrow do  
      TMatrix[lrow,lcolumn] := KineticIntegral(lrow,lcolumn);  
      TMatrix[lcolumn,lrow] := TMatrix[lrow,lcolumn];  
    end do:  
  end do:  
  print(TMatrix);
```

$$\begin{bmatrix} 2.002500000 & 0.2919625292 \\ 0.2919625292 & 0.3027000000 \end{bmatrix}$$

(16.4.2.1)

Potential Energy Matrix V

```
> VMatrix := array(1..Dim,1..Dim):  
  for lrow to Dim do  
    for lcolumn to lrow do  
      VMatrix[lrow,lcolumn] := PotentialIntegral(lrow,lcolumn);  
      VMatrix[lcolumn,lrow] := VMatrix[lrow,lcolumn];  
    end do:  
  end do:  
  print(VMatrix);
```

$$\begin{bmatrix} -1.843786751 & -0.7765769962 \\ -0.7765769962 & -0.7168538764 \end{bmatrix}$$

(16.4.3.1)

solution of the general eigenvalue problem

Orthogonalisation of the basis set

The general eigenvalue problem can be simplified to the special eigenvalue problem by a transformation to an orthonormal basis set, which is solved by diagonalization. Orthonormalization means that the overlap matrix has to be transformed to the unity matrix, i.e. transformation matrix **A** must be found such that $\mathbf{A}^t \mathbf{S} \mathbf{A} = \mathbf{I}$. To this purpose we first diagonalize $\mathbf{B}^t \mathbf{S} \mathbf{B} = \mathbf{D}$, in which **B** = "EigenVectors" is the transformation matrix, and **D** = "SSpur" is diagonal.

```
> SSpur := diag(eigenvals(SMatrix));
```

$$SSpur := \begin{bmatrix} 0.444836651500004 & 0 \\ 0 & 1.55516334850000 \end{bmatrix} \quad (16.5.1.1)$$

```
> evalf(Diagonalisation(SMatrix,EigenValues,EigenVectors));
```

$$\begin{bmatrix} 0.4448366515 & 1.555163348 \end{bmatrix} \quad (16.5.1.2)$$

```
> print(EigenVectors);
```

$$\begin{bmatrix} 0.7071067810 & 0.7071067814 \\ -0.7071067814 & 0.7071067810 \end{bmatrix} \quad (16.5.1.3)$$

Then we form the square-root of the diagonal matrix = "RootSMatrix"

```
> RootSMatrix := SSpur:
```

```
> for i to Dim do
```

```
    RootSMatrix[i,i] := sqrt(EigenValues[i]);
```

```
end do;
```

```
print(RootSMatrix);
```

$$\begin{bmatrix} 0.6669607571 & 0 \\ 0 & 1.247061886 \end{bmatrix} \quad (16.5.1.4)$$

Multiplying the "inverse RootSMatrix" W with the EigenVectors B from the left yields the desired transformation matrix: $\mathbf{A} = \mathbf{B}\mathbf{W}$, as $(\mathbf{B}\mathbf{W})^t \mathbf{S} \mathbf{B}\mathbf{W} = \mathbf{W}(\mathbf{B}^t \mathbf{S} \mathbf{B}) \mathbf{W} = \mathbf{W}\mathbf{D}\mathbf{W} = \mathbf{I}$

```
> TransMatrix := evalm(EigenVectors &* inverse(RootSMatrix));
```

$$TransMatrix := \begin{bmatrix} 1.060192483 & 0.5670181965 \\ -1.060192483 & 0.5670181961 \end{bmatrix} \quad (16.5.1.5)$$

Hamilton Operator, transformation to the orthogonal basis-set and diagonalisation

The electronic Hamilton operator in a one electron problem is $\mathbf{H} = \mathbf{H}_{core} = \mathbf{T} + \mathbf{V} = \mathbf{T}\mathbf{V}\mathbf{Matrix}$.

```
> TVMatrix := evalm( TMatrix &+ VMatrix); Hcore = T + V
```

$$TVMatrix := \begin{bmatrix} 0.158713249 & -0.4846144670 \\ -0.4846144670 & -0.4141538764 \end{bmatrix} \quad (16.5.2.1)$$

Transformation to the orthonormal basis-set yields $\mathbf{H}' = \mathbf{A}^t \mathbf{H} \mathbf{A} = \mathbf{A}^t \mathbf{T}\mathbf{V}\mathbf{Matrix} \mathbf{A}$.

```
> HamiltonOp := evalm(transpose(TransMatrix) &* TVMatrix &*  
    TransMatrix);
```

$$HamiltonOp := \begin{bmatrix} 0.8023038390 & 0.3443781728 \\ 0.3443781729 & -0.3937430635 \end{bmatrix} \quad (16.5.2.2)$$

The Hamilton operator H' is diagonalized and yields as eigenvalues the orbital energies and as eigenvectors coefficients c' representing the molecular orbitals in the orthonormal basis set

```
> evalf(Diagonalisation(HamiltonOp,EigenValues,OBEigenVec));
```

$$\begin{bmatrix} -0.4858126308 & 0.8943734063 \end{bmatrix} \quad (16.5.2.3)$$

```
> print(OBEigenVec);
```

$$\begin{bmatrix} -0.2582790833 & 0.9660703469 \\ 0.9660703469 & 0.2582790833 \end{bmatrix} \quad (16.5.2.4)$$

Back-transformation of the coefficients to the original basis set: $c = Ac'$

```
> MOEigenVectors := evalm(TransMatrix &* OBEigenVec);
```

$$MOEigenVectors := \begin{bmatrix} 0.2739539232 & 1.170669460 \\ 0.8216050080 & -0.8777715801 \end{bmatrix} \quad (16.5.3.1)$$

3.83 procedures for small molecules

Procedures for calculation *ab initio* for small molecules utilizing basis sets comprising only s functions (spherical gaussians)

copyright P. Vogt and Hanspeter Huber @ 1999 December

modified by J. F. Ogilvie and G. J. Fee 2006 October 16

This section group contains three programs from Professor Huber -- SCF program for small systems, MP2 program for small systems and CI program for small systems.

All properties have atomic units!

The following conversion factors are applicable between atomic units and SI units.

For energy, $1 \text{ a.u.} = \frac{e^2}{a_0} = 1 \text{ hartree}$ or $1 E_h = 2 \text{ rydberg} = 2 * 10973731.568525 \text{ m}^{-1}$

$$= 2 * 13.6056923 \text{ eV} = 27.2113845 \text{ eV} = 4.35974417 \cdot 10^{-18} \text{ J};$$

for length, $1 \text{ a.u.} = a_0 = 0.5291772108 \cdot 10^{-10} \text{ m};$

for charge, $1 \text{ a.u.} = |e| = 1.60217653 \cdot 10^{-19} \text{ C};$

for mass, $1 \text{ a.u.} = m_e = 9.1093826 \cdot 10^{-31} \text{ kg} = 5.4857990945 \cdot 10^{-4} \text{ u};$

for action or angular momentum, $1 \text{ a.u.} = \frac{h}{2\pi} = 1.05457168 \cdot 10^{-34} \text{ J s},$ and

for electric dipolar moment, $1 \text{ a.u.} = |e| a_0 = 8.47835309 \cdot 10^{-30} \text{ C m}.$

synopsis of procedures

These procedures are similar in design and scope to those in the one-electron program for small molecules.

input of data for nuclear and basis set and occupation matrix

```
> restart:
```

```
> with(LinearAlgebra):
```

```
> interface( rtablesize = 15 );
```

We set the numerical precision to 16 decimal digits.

```
> Digits := 16;
```

Digits := 16 (17.3.1)

This suite contains as default content appropriate to lithium hydride, LiH, hence involving two atomic centres.

```
> NumberofNuclei := 2;
```

NumberofNuclei := 2 (17.3.2)

We define the nuclear data through atomic numbers, for H, $Z=1$, and for Li, $Z=3$.

```
> AtomicNumber := <1,3>;
```

AtomicNumber := $\begin{bmatrix} 1 \\ 3 \end{bmatrix}$ (17.3.3)

We define the locations of the atomic centres, placing the atomic nucleus of H at the origin and the nucleus of Li $3 a_0$ away along axis x . For convenience of application of other internuclear separations, we specify a variable aa to which we here assign a value 3.

```
> aa := 3;
```

aa := 3 (17.3.4)

```
> NuclearCoordinate := <<0|0|0>,<aa|0|0>>;
```

NuclearCoordinate := $\begin{bmatrix} 0 & 0 & 0 \\ 3 & 0 & 0 \end{bmatrix}$ (17.3.5)

We define data for the basis set comprising ten exponents α , four basis functions of H and three each in two sets for Li.

```
> Dim := 10:
```

```
> Exponent := <13.36|2.0133|0.4538|0.1233|28.63|4.483|1.303|4.869|  
0.8569|0.0635>; # |0.2433
```

Exponent := (17.3.6)

$\begin{bmatrix} 13.36 & 2.0133 & 0.4538 & 0.1233 & 28.63 & 4.483 & 1.303 & 4.869 & 0.8569 & 0.0635 \end{bmatrix}$

We define the locations of spherical gaussian functions, which need not be identical with locations of atomic nuclei.

```
> BasisCoordinate := <seq(<0|0|0>,i=1..4), seq(<aa|0|0>,i=5..Dim)>;
```

$$BasisCoordinate := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 3 & 0 & 0 \\ 3 & 0 & 0 \\ 3 & 0 & 0 \\ 3 & 0 & 0 \\ 3 & 0 & 0 \\ 3 & 0 & 0 \end{bmatrix} \quad (17.3.7)$$

For the occupation matrix in a molecular-orbital representation, or 'MO basis', we have the first two molecular orbitals occupied with two electrons each.

```
> Occupation := Matrix([[2,0,0,0,0,0,0,0,0,0],[0,2,0,0,0,0,0,0,0,0],
    seq([0,0,0,0,0,0,0,0,0,0], j=1..8)], datatype=float[8]
);
```

$$Occupation := \begin{bmatrix} 2. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 2. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{bmatrix} \quad (17.3.8)$$

parameters and auxiliary functions

function to calculate the square of the distance between two basis functions

```
> #X := 1:
    #Y := 2:
    #Z := 3:
```

Although use of these values as indices would make the program more readable, as x , y and z -coordinates are stored in arrays, their use also prevents evalhf from operating efficiently; for this reason we apply only numerical indices but with the same meaning and purpose.

```
> sqrDistance := proc (func1, func2)
    (BasisCoordinate[func1,1] - BasisCoordinate[func2,1])^2 +
```

```

    (BasisCoordinate[func1,2] - BasisCoordinate[func2,2])^2 +
    (BasisCoordinate[func1,3] - BasisCoordinate[func2,3])^2;
end proc:

```

calculation of the product of two gaussian functions

The product of two gaussian functions is also a gaussian function with a new exponent c_{new} , which is the sum of the original exponents, and new coordinates $R_{xnew}=R_1$, $R_{ynew}=R_2$, $R_{znew}=R_3$, which are found from the old ones by weighting with the exponents.

```

> ProductGaussian := proc(func1,func2,R)
    local a,b;
    R[4] := Exponent[func1] + Exponent[func2];
    a := Exponent[func1] / R[4];
    b := Exponent[func2] / R[4];
    R[1] := a*BasisCoordinate[func1,1] + b*BasisCoordinate[func2,1];
    R[2] := a*BasisCoordinate[func1,2] + b*BasisCoordinate[func2,2];
    R[3] := a*BasisCoordinate[func1,3] + b*BasisCoordinate[func2,3];
end proc:

```

auxiliary function for the calculation of the potential and the two-electron-integrals

```

> rootPi := Pi^(1/2):
AuxInt := proc(X)
    local rootX;
    if X=0
    then 1
    else
    rootX := X^(1/2);
    evalf(1/2*rootPi/rootX*erf(rootX));
    end if;
end proc:

```

function to form and to sort eigenvalues and eigenvectors of a real matrix made symmetric

A matrix to be diagonalized is input, and eigenvalues and eigenvectors of the symmetrized matrix are returned sorted in ascending order.

```

> Seig := proc(A::Matrix)
    local A1,evalu,evect,evalul,ss,ev2,ev3,ev4,
           evecl,evecl1,evecl2,evecM;
    A1 := 1/2*(A + LinearAlgebra:-Transpose(A));
    evalu, evect := evalf(LinearAlgebra:-Eigenvectors(A1,
        output=['values', 'vectors']));
    evalul := [seq([Re(evalu[i]), i], i=1..RowDimension(A1))];
    ss := proc(a,b) evalb(a[1] < b[1]); end proc;
    ev2 := sort(evalul, ss);
    ev3 := map2(op, 1, ev2);
    ev3 := [seq(op(1, ev2[j]), j=1..RowDimension(A1))];
    ev4 := map2(op, 2, ev2);

```

```

ev4 := [seq(op(2, ev2[j]), j=1..RowDimension(A1))];
evecl1 := convert(
    LinearAlgebra:-Transpose(map(Re, evec)), listlist);
evecl := [seq(evecl1[ev4[i]], i=1..RowDimension(A1))];
evecl2 := map(proc(a) map(Re, a) end proc, evecl);
evecM := LinearAlgebra:-Transpose(Matrix(evecl2));
convert(ev3, Vector), evecM;
end proc:
Warning, (in Seig) `i` is implicitly declared local
Warning, (in Seig) `j` is implicitly declared local

```

integral functions -- S, T, V, two-electron integrals, M

function to calculate the overlap integrals $\langle j|j \rangle$

```

> Overlapintegral := proc (func1, func2)
    local alpha, beta, cinv, Q, aux;
    alpha := Exponent[func1];
    beta := Exponent[func2];
    cinv := 1/(alpha + beta);
    Q := exp(-alpha * beta * cinv * sqrDistance(func1,func2));
    aux := (4*alpha*beta*cinv^2);
    Q*aux^(3/4);
end proc:

```

function to calculate the dipole integrals $\langle j|x|j \rangle$

The electric dipolar moment is assumed to lie along only axis x, with label 1 in BasisCoordinates below.

```

> DipoleIntegral := proc (func1, func2)
    local alpha, beta, cinv, Q, aux;
    alpha := Exponent[func1];
    beta := Exponent[func2];
    cinv := 1/(alpha + beta);
    Q := exp(-alpha * beta * cinv * sqrDistance(func1,func2));
    aux := (4*alpha*beta*cinv^2);
    Q*aux^(3/4)*(alpha*BasisCoordinate[func1,1]
        +beta*BasisCoordinate[func2,1])*cinv;
end proc:

```

function to calculate the kinetic-energy integral $\langle j|T|j \rangle$

```

> KineticIntegral := proc (func1, func2)
    local alpha, beta, E;
    alpha := Exponent[func1];
    beta := Exponent[func2];
    E := alpha * beta /(alpha+beta);
    Overlapintegral(func1,func2)*E*(3-2*E*sqrDistance(func1,func2));
end proc:

```

function to calculate the potential-energy integral $\langle j|V|j\rangle$

```
> R := Vector(4, datatype=float[8]):
> PotentialIntegral := proc(func1, func2)
    local V, argument, nuc;
    global NumberofNuclei, R;
    ProductGaussian(func1, func2, R);
    V := 0;
    for nuc to NumberofNuclei do
        argument := R[4]*((R[1] - NuclearCoordinate[nuc,1])^2 +
                           (R[2] - NuclearCoordinate[nuc,2])^2 +
                           (R[3] - NuclearCoordinate[nuc,3])^2);
        V := V + AtomicNumber[nuc] * AuxInt(argument);
    end do;
    evalf(-Overlapintegral(func1, func2) * 2/Pi^(1/2) * R[4]^(1/2) *
V);
end proc:
```

function to calculate the two-electron integral $\langle jj|1/r|jj\rangle$

```
> R1 := Vector(4, datatype=float[8]):
R2 := Vector(4, datatype=float[8]):
> TwoElectronIntegral := proc(i, j, k, l)
    local Argument, cnew;
    global R1, R2;
    ProductGaussian(i, j, R1);
    ProductGaussian(k, l, R2);
    cnew := R1[4] * R2[4] / (R1[4] + R2[4]);
    Argument := cnew*((R1[1]-R2[1])^2 + (R1[2]-R2[2])^2 +
                      (R1[3]-R2[3])^2);
    evalf(2/Pi^(1/2)) * Overlapintegral(i, j) * Overlapintegral(k,
l)
        * cnew^(1/2) * AuxInt(Argument);
end proc:
```

MOperator - 2-e-Operator for closed shells

```
> MOperator := proc(i, j)
    local sumd, k, l;
    global Dim;
    sumd := 0;
    for k to Dim do
        for l to Dim do
            sumd := sumd + DensityMatrix[k, l]*(TwoElectronIntegral(i, j,
k, l)
                - 1/2*TwoElectronIntegral(i, k, j, l));
        end do;
    end do;
```

```

    end do;
    sumd;
end proc:
construction of integral matrices
overlap matrix S
> SMatrix := Matrix(Dim, datatype=float[8]):
makeSMatrix := proc(SMatrix,Dim)
    local ro,colm;
    for ro to Dim do
        for colm to ro do
            SMatrix[ro,colm] := Overlapintegral(ro,colm);
            SMatrix[colm,ro] := SMatrix[ro,colm];
        end do:
    end do:
end proc:
evalhf(makeSMatrix(SMatrix,Dim)):
> SMatrix := Matrix(SMatrix, shape=symmetric, datatype=float[8]);
SMatrix := [[ 1.00000000000000022, 0.554217906990995046, 0.212852355178614228,
0.0830668703531271951, 2.23347194687782627 × 10-36, 6.12865672928848465
× 10-14, 9.82607014933355889 × 10-6, 9.38406416912951933 × 10-15,
0.000233852732252874028, 0.0287843411454523773 ],
[ 0.554217906990995046, 0.99999999999999989, 0.682094585947494614,
0.318503225639132193, 1.54997416854734232 × 10-8, 3.30123760638162570
× 10-6, 0.000781268911718861108, 2.35069338573332284 × 10-6,
0.00391674745688022630, 0.116105892687594928 ],
[ 0.212852355178614228, 0.682094585947494614, 1.00000000000000022,
0.742230947050620871, 0.00221443639637228544, 0.0107647051521392763,
0.0396049869395664819, 0.00995496377162429988, 0.0642701709697378332,
0.322053326568992826 ],
[ 0.0830668703531271951, 0.318503225639132193, 0.742230947050620871, 1.,
0.0156488282504379864, 0.0622843974740923262, 0.152892046033300560,
0.0585952510033894147, 0.204729013192892800, 0.632346045797274825 ],
[ 2.23347194687782627 × 10-36, 1.54997416854734232 × 10-8,
0.00221443639637228544, 0.0156488282504379864, 1., 0.566030410892383373,
0.260701750799630427, 0.591824678947436733, 0.194721961726699117,

```

(17.6.1.1)


```

0.0288115092752221288 ],
[ 6.12865672928848465 × 10-14, 3.30123760638162570 × 10-6,
0.0107647051521392763, 0.0622843974740923262, 0.566030410892383373, 1.,
0.763593486998213611, 0.998722033661039954, 0.628954847417687479,
0.113706698134348860 ],
[ 9.82607014933355889 × 10-6, 0.000781268911718861108,
0.0396049869395664819, 0.152892046033300560, 0.260701750799630427,
0.763593486998213611, 1., 0.737384934170873141, 0.967833043769922408,
0.273161283839884139 ],
[ 9.38406416912951933 × 10-15, 2.35069338573332284 × 10-6,
0.00995496377162429988, 0.0585952510033894147, 0.591824678947436733,
0.998722033661039954, 0.737384934170873141, 1.00000000000000022,
0.602638388384419410, 0.107054389086686977 ],
[ 0.000233852732252874028, 0.00391674745688022630, 0.0642701709697378332,
0.204729013192892800, 0.194721961726699117, 0.628954847417687479,
0.967833043769922408, 0.602638388384419410, 1.00000000000000022,
0.360875982764080394 ],
[ 0.0287843411454523773, 0.116105892687594928, 0.322053326568992826,
0.632346045797274825, 0.0288115092752221288, 0.113706698134348860,
0.273161283839884139, 0.107054389086686977, 0.360875982764080394,
1.00000000000000022 ]]

```

dipole matrix D

```

> DMatrix := Matrix(Dim, datatype=float[8]):
makeDMatrix := proc(DMatrix,Dim)
    local ro,colm;
    for ro to Dim do
        for colm to ro do
            DMatrix[ro,colm] := DipoleIntegral(ro,colm);
            DMatrix[colm,ro] := DMatrix[ro,colm];
        end do:
    end do:
end proc:
evalhf(makeDMatrix(DMatrix,Dim)):
DMatrix := Matrix(DMatrix, shape=symmetric, datatype=float[8]):

```

> DMatrix;

```
[ [ 0., 0., 0., 0.,  $4.56853787847907853 \times 10^{-36}$ ,  $4.61941962406550624 \times 10^{-14}$ ,  
2.61952589604752617  $\times 10^{-6}$ ,  $7.51950328150062507 \times 10^{-15}$ ,  
0.0000422852533817121362, 0.000408493834559442610 ],  
[ 0., 0., 0., 0.,  $4.34441725716653389 \times 10^{-8}$ ,  $6.83440490251781448 \times 10^{-6}$ ,  
0.000920899851011376474,  $4.98911385516566290 \times 10^{-6}$ ,  
0.00350804218779248757, 0.0106501216087186226 ],  
[ 0., 0., 0., 0., 0.00653965238670378701, 0.0293255792398154913,  
0.0881238011992061532, 0.0273187337138566452, 0.126054267576031942,  
0.118598798978142544 ],  
[ 0., 0., 0., 0., 0.0467451686738606931, 0.181851564515786546,  
0.419024754921245079, 0.171444190334416824, 0.536928049596989920,  
0.644871101308248651 ],  
[  $4.56853787847907853 \times 10^{-36}$ ,  $4.34441725716653389 \times 10^{-8}$ ,  
0.00653965238670378701, 0.0467451686738606931, 3., 1.69809123267715023,  
0.782105252398891393, 1.77547403684231031, 0.584165885180097377,  
0.0864345278256663935 ],  
[  $4.61941962406550624 \times 10^{-14}$ ,  $6.83440490251781448 \times 10^{-6}$ ,  
0.0293255792398154913, 0.181851564515786546, 1.69809123267715023,  
2.99999999999999956, 2.29078046099464094, 2.99616610098311931,  
1.88686454225306188, 0.341120094403046525 ],  
[  $2.61952589604752617 \times 10^{-6}$ , 0.000920899851011376474,  
0.0881238011992061532, 0.419024754921245079, 0.782105252398891393,  
2.29078046099464094, 3., 2.21215480251261942, 2.90349913130976711,  
0.819483851519652529 ],  
[  $7.51950328150062507 \times 10^{-15}$ ,  $4.98911385516566290 \times 10^{-6}$ ,  
0.0273187337138566452, 0.171444190334416824, 1.77547403684231031,  
2.99616610098311931, 2.21215480251261942, 3.000000000000000089,  
1.80791516515325812, 0.321163167260060889 ],  
[ 0.0000422852533817121362, 0.00350804218779248757, 0.126054267576031942,  
0.536928049596989920, 0.584165885180097377, 1.88686454225306188,
```

(17.6.2.1)

```

2.90349913130976711, 1.80791516515325812, 3.000000000000000089,
1.08262794829224140 ],
[ 0.000408493834559442610, 0.0106501216087186226, 0.118598798978142544,
0.644871101308248651, 0.0864345278256663935, 0.341120094403046525,
0.819483851519652529, 0.321163167260060889, 1.08262794829224140,
3.000000000000000089 ]]

```

kinetic-energy matrix T

```

> TMatrix := Matrix(Dim, datatype=float[8]):
makeTMatrix := proc(TMatrix,Dim)
    local ro,colm;
    for ro to Dim do
        for colm to ro do
            TMatrix[ro,colm] := KineticIntegral(ro,colm);
            TMatrix[colm,ro] := TMatrix[ro,colm];
        end do;
    end do;
end proc;
evalhf(makeTMatrix(TMatrix, Dim)):
TMatrix := Matrix(TMatrix, shape=symmetric, datatype=float[8]):
> TMatrix;
[[ 20.04000000000000063, 2.90903976626816663, 0.280257665747629892,
0.0304454529818951325,  $-3.27489388180811669 \times 10^{-33}$ ,  $-1.18123108032241435$ 
 $\times 10^{-11}$ ,  $-0.000214295192642665526$ ,  $-2.05048911112441679 \times 10^{-12}$ ,
 $-0.00216453909576828874$ , 0.00338801479724379958 ],
[ 2.90903976626816663, 3.01994999999999969, 0.757795616752319323,
0.111015453099274589,  $-8.99689377128428771 \times 10^{-7}$ ,
 $-0.000100942455882715147$ ,  $-0.00694572078669267179$ ,
 $-0.0000757970217198834529$ ,  $-0.0184085537815467694$ , 0.0135223152284662182
],
[ 0.280257665747629892, 0.757795616752319323, 0.6807000000000000194,
0.215892127802988054,  $-0.00498666306711393909$ ,  $-0.0195961537797276417$ ,
 $-0.0407692773585963397$ ,  $-0.0184801373730618772$ ,  $-0.0446239675876296815$ ,
0.0358317666540269725 ],
[ 0.0304454529818951325, 0.111015453099274589, 0.215892127802988054,

```

(17.6.3.1)

0.184950000000000003, 0.00151799412379349341, 0.00627830474694125989,
0.0167476512020009356, 0.00588663040717223984, 0.0233869476423997705,
0.0595164205867178275],
[$-3.27489388180811669 \times 10^{-33}$, $-8.99689377128428771 \times 10^{-7}$,
 -0.00498666306711393909 , 0.00151799412379349341, 42.9450000000000003,
6.58191966835092934, 0.974721892532084078, 7.38828441847752160,
0.486024950636381958, 0.00547644601598622043],
[$-1.18123108032241435 \times 10^{-11}$, -0.000100942455882715147 ,
 -0.0195961537797276417 , 0.00627830474694125989, 6.58191966835092934,
6.72449999999999903, 2.31269411597832431, 6.99310261964435043,
1.35739573705660366, 0.0213585896478087484],
[-0.000214295192642665526 , -0.00694572078669267179 ,
 -0.0407692773585963397 , 0.0167476512020009356, 0.974721892532084078,
2.31269411597832431, 1.9544999999999990, 2.27391270231115161,
1.50093752142321346, 0.0496191025368911895],
[$-2.05048911112441679 \times 10^{-12}$, $-0.0000757970217198834529$,
 -0.0184801373730618772 , 0.00588663040717223984, 7.38828441847752160,
6.99310261964435043, 2.27391270231115161, 7.303500000000000144,
1.31735919190721762, 0.0201313147082040622],
[-0.00216453909576828874 , -0.0184085537815467694 ,
 -0.0446239675876296815 , 0.0233869476423997705, 0.486024950636381958,
1.35739573705660366, 1.50093752142321346, 1.31735919190721762,
1.285350000000000021, 0.0640039080232702728],
[0.00338801479724379958, 0.0135223152284662182, 0.0358317666540269725,
0.0595164205867178275, 0.00547644601598622043, 0.0213585896478087484,
0.0496191025368911895, 0.0201313147082040622, 0.0640039080232702728,
0.09525000000000000291]]

potential-energy matrix V

```
> VMatrix := Matrix(Dim, datatype=float[8]):
  makeVMatrix := proc(VMatrix,Dim)
    local ro,colm;
    for ro to Dim do
```

```

    for colm to ro do
      VMatrix[ro,colm] := PotentialIntegral(ro,colm);
      VMatrix[colm,ro] := VMatrix[ro,colm];
    end do:
  end do:
end proc:
evalhf(makeVMatrix(VMatrix,Dim)):
VMatrix := Matrix(VMatrix, shape=symmetric, datatype=float[8]):
> VMatrix;
[[ -6.83274897756825794, -3.00621055871727627, -1.10552063175511184,
  -0.427243205870134091, -8.11162436709966410 × 10-36, -1.63160821419444342
  × 10-13, -0.0000421571881786969390, -2.45150022336215955 × 10-14,
  -0.00110895547973680216, -0.147813255488461498 ],
[ -3.00621055871727627, -3.26424962202284652, -1.89100199242431266,
  -0.843831173475865670, -2.12467594488842781 × 10-7,
  -0.0000122381144780725297, -0.00194811828574735111, -9.13413342917451797
  × 10-6, -0.00981744232329287252, -0.307475956429912656 ],
[ -1.10552063175511184, -1.89100199242431266, -2.07493172490416145,
  -1.37752749585652312, -0.0403336747874433119, -0.0758311196766028872,
  -0.148684609630984232, -0.0732205911193773151, -0.201155612363712294,
  -0.619773574658817328 ],
[ -0.427243205870134091, -0.843831173475865670, -1.37752749585652312,
  -1.52520902878749265, -0.288843518137547495, -0.469405286512686815,
  -0.654688457939976098, -0.459199648593019538, -0.733551882263327903,
  -1.03088874426586674 ],
[ -8.11162436709966410 × 10-36, -2.12467594488842781 × 10-7,
  -0.0403336747874433119, -0.288843518137547495, -25.9487831045908912,
  -11.2146096628422995, -4.91521323630380724, -11.7926634598802345,
  -3.64427085015619268, -0.532041028138585781 ],
[ -1.63160821419444342 × 10-13, -0.0000122381144780725297,
  -0.0758311196766028872, -0.469405286512686815, -11.2146096628422995,
  -10.4695452402963678, -6.47220224778489062, -10.6717801978576183,
  -5.12962037062709264, -0.858633464529843793 ],

```

(17.6.4.1)

```
[ -0.0000421571881786969390, -0.00194811828574735111,
-0.148684609630984232, -0.654688457939976098, -4.91521323630380724,
-6.47220224778489062, -5.79799803322036311, -6.44710627471902953,
-5.13758097467689367, -1.17199043423317462 ],
[ -2.45150022336215955 × 10-14, -9.13413342917451797 × 10-6,
-0.0732205911193773151, -0.459199648593019538, -11.7926634598802345,
-10.6717801978576183, -6.44710627471902953, -10.8969151207612693,
-5.08239849118629916, -0.840533666068981500 ],
[ -0.00110895547973680216, -0.00981744232329287252,
-0.201155612363712294, -0.733551882263327903, -3.64427085015619268,
-5.12962037062709264, -5.13758097467689367, -5.08239849118629916,
-4.76489076875933470, -1.29227281750674461 ],
[ -0.147813255488461498, -0.307475956429912656, -0.619773574658817328,
-1.03088874426586674, -0.532041028138585781, -0.858633464529843793,
-1.17199043423317462, -0.840533666068981500, -1.29227281750674461,
-1.49618146841245259 ]]
```

SCF step -- solution of the general eigenvalue problem

orthogonalisation of the basis set

A general eigenvalue problem becomes simplified to a special eigenvalue problem on a transformation to an orthonormal basis set; this result is solved by diagonalization. Othonormalization means that the overlap matrix must be transformed to a unit matrix; i.e. a transformation matrix **A** has to be found such that $A^T \mathbf{S} \mathbf{A} = \mathbf{I}$. For this purpose we first diagonalize $B^T \mathbf{S} \mathbf{B} = \mathbf{D}$, in which **B** = "EigenVectors" is the transformation matrix, and **D** = "SSpur" is diagonal.

```
> SSpur := simplify(DiagonalMatrix(Eigenvalues(SMatrix))) ;
SSpur := [[ 0.0001460384808802236, 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0.01141397601751697, 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0.09605619463081899, 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0.2986898301303021, 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.3411055824838217, 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0.6401163397349950, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0.9732667622186892, 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 1.380738789499534, 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 2.522050980499614, 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 3.736415506303832 ]]
```

(17.7.1.1)

```
> SSpur := Matrix(Dim, datatype=float[8]):
> ESSp := Eigenvalues(SMatrix);
```

$$ESSp := \begin{bmatrix} 0.0001460384808802236 \\ 0.01141397601751697 \\ 0.09605619463081899 \\ 0.2986898301303021 \\ 0.3411055824838217 \\ 0.6401163397349950 \\ 0.9732667622186892 \\ 1.380738789499534 \\ 2.522050980499614 \\ 3.736415506303832 \end{bmatrix}$$

(17.7.1.2)

```
> makeSSpur := proc(SSpur, Dim, ESSp)
  local i;
  for i to Dim do
    SSpur[i,i] := ESSp[i];
  end do;
end proc;
> evalhf(makeSSpur(SSpur, Dim, ESSp)):
> EValues,EVectors := evalf(Seig(SMatrix));
```

$$EValues, EVectors := \begin{bmatrix} 0.0001460384808800710 \\ 0.01141397601751448 \\ 0.09605619463081945 \\ 0.2986898301303011 \\ 0.3411055824838211 \\ 0.6401163397349954 \\ 0.9732667622186881 \\ 1.380738789499524 \\ 2.522050980499614 \\ 3.736415506303813 \end{bmatrix}, \begin{bmatrix} 0.0001204891586674577, \\ 0.002366839799075695, 0.1494183214115198, -0.3123234990585622, \\ 0.2486842657016717, -0.5310693862983760, -0.3979773653347594, \\ -0.5465656701239350, -0.2799564144711800, -0.02735716657136172], \\ [-0.0005445290270625364, -0.008798316234178189, -0.4290308956246858, \\ 0.4814451511486337, -0.3350363370462271, 0.2146256883002170, \end{bmatrix}$$

(17.7.1.3)

```
0.002366839799075695, 0.1494183214115198, -0.3123234990585622,
0.2486842657016717, -0.5310693862983760, -0.3979773653347594,
-0.5465656701239350, -0.2799564144711800, -0.02735716657136172 ],
[-0.0005445290270625364, -0.008798316234178189, -0.4290308956246858,
0.4814451511486337, -0.3350363370462271, 0.2146256883002170,
```

—0.1000876937444087, —0.4363336582563866, —0.4709210404170322,
—0.05983797173120276],
[0.0007867946820797423, 0.009453529154392664, 0.6711390217030118,
—0.1243865465277918, —0.006910417970065845, 0.4220670806658931,
0.2151560202071073, —0.03066074322588277, —0.5454618266950732,
—0.1051724264466175],
[—0.0006220500089867232, —0.008817562123946003, —0.5513035144806810,
—0.4263817044733030, 0.3075806163053168, —0.006876142760372255,
0.2394381026937365, 0.3205450439831252, —0.4826499017378594,
—0.1627917869745268],
[—0.01347176071267742, 0.04971982807706140, —0.005689116780593268,
—0.2507413399751962, —0.5061016015359729, —0.3230726875458869,
0.6325499060659480, —0.2858532027655441, 0.1262332065147427,
—0.2759742969901979],
[—0.7186624296608934, —0.005873562537498435, 0.006876504233422381,
0.2140547640208790, 0.3862804945638905, 0.07796735991978562,
0.1112162587233649, —0.1586349901595659, 0.1571873254332190,
—0.4691370629879192],
[0.1181454854174152, 0.7462127052025919, —0.02002851940959909,
—0.1457383691370257, —0.1713276696769817, 0.1415538910425516,
—0.3511642086778225, 0.1041560808518843, 0.07900367816324055,
—0.4666587433842398],
[0.6813899624092389, —0.2008104578133091, 0.009431284154407382,
0.2201471294189705, 0.3882227906984180, 0.06162352017768451,
0.1474255827241010, —0.1727467134819538, 0.1590284909603563,
—0.4641648280452598],
[—0.07129478186492947, —0.6306587242757820, 0.01220278359430709,
—0.2302432776486303, —0.3744116043957120, 0.07736471458428872,
—0.4124363221870925, 0.1925781734331447, 0.03155567415619921,
—0.4355118492778933],
[0.002815016495350880, 0.04852464493959572, 0.1962774905500127,
0.4919188634312651, —0.07515046752485452, —0.5945065911452674,

0.01819309109668673, 0.4717375518261540, −0.3101744142964507,
−0.1974012656605701]]

We form the square root of the diagonal matrix = 'RootSMatrix'.

```
> RootSMatrix := copy(SSpur);
RootSMatrix := [[ 0.000146038480880223608, 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0.0114139760175169698, 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0.0960561946308189918, 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0.298689830130302092, 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.341105582483821723, 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0.640116339734995043, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0.973266762218689241, 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 1.38073878949953399, 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 2.52205098049961407, 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 3.73641550630383179 ]]
```

(17.7.1.4)

```
> makeRootSMatrix := proc(RootSMatrix,Dim,EValues)
    local i;
    for i to Dim do
        RootSMatrix[i,i] := sqrt(EValues[i]);
    end do;
end proc;
```

```
> evalhf(makeRootSMatrix(RootSMatrix,Dim,EValues));
```

```
> RootSMatrix;
[[ 0.0120846382188326603, 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0.106836211171655093, 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0.309929338125353115, 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0.546525232839528696, 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.584042449213942882, 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0.800072709030245188, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0.986542833443479039, 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 1.17504842006596655, 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 1.58809665338719674, 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 1.93298098963849441 ]]
```

(17.7.1.5)

Multiplying the 'inverse RootSMatrix' **W** with the EigenVectors **B** from the left yields the desired transformation matrix:

$$\mathbf{A} = \mathbf{B} \mathbf{W}, \text{ as } (\mathbf{B}\mathbf{W})^T \mathbf{S} \mathbf{B} \mathbf{W} = \mathbf{W} (\mathbf{B}^T \mathbf{S} \mathbf{B}) \mathbf{W} = \mathbf{W} \mathbf{D} \mathbf{W} = \mathbf{I}$$

> TransMatrix := EVectors.MatrixInverse(RootSMatrix);

TransMatrix := [[0.009970439866349312, 0.02215390992547333, 0.4821044768310592, (17.7.1.6)

−0.5714713251863102, 0.4257982720885674, −0.6637764046996132,
−0.4034060679814978, −0.4651431045652152, −0.1762842418149239,
−0.01415283787994110],
[−0.04505960519479550, −0.08235331576895611, −1.384286167355868,
0.8809202617182657, −0.5736506610044345, 0.2682577294260683,
−0.1014529631674051, −0.3713324921809528, −0.2965317252023803,
−0.03095631672114564],
[0.06510701171455874, 0.08848618881854167, 2.165458183992781,
−0.2275952491370409, −0.01183204744683629, 0.5275359050522715,
0.2180909058516151, −0.02609317429162747, −0.3434689101143034,
−0.05440944686491037],
[−0.05147444199176129, −0.08253345965048043, −1.778803897092512,
−0.7801683780600440, 0.5266408575597316, −0.008594397337595370,
0.2427042137217598, 0.2727930513409225, −0.3039172085077013,
−0.08421799689037403],
[−1.114783948739406, 0.4653836703098346, −0.01835617374916687,
−0.4587918817077182, −0.8665493445161911, −0.4038041591713307,
0.6411783499131649, −0.2432692967235311, 0.07948710567804813,
−0.1427713456415371],
[−59.46908932208928, −0.05497726354280111, 0.02218732913449172,
0.3916649244331048, 0.6613911284766746, 0.09745034299981129,
0.1127333299205774, −0.1350029389858336, 0.09897843755161849,
−0.2427013330719084],
[9.776501644318790, 6.984642164103353, −0.06462285736079111,
−0.2666635689990503, −0.2933479748048622, 0.1769262836300549,
−0.3559543455929848, 0.08863982034547733, 0.04974739918677891,
−0.2414192099589734],
[56.38480441618542, −1.879610439298192, 0.03043043363191655,
0.4028123793574419, 0.6647167362936092, 0.07702239994209694,
0.1494365756117444, −0.1470124213879253, 0.1001377911232102,
−0.2401290186159916],

```
[ -5.899620706379436, -5.903042773226903, 0.03937279274081368,
-0.4212857226232308, -0.6410691635507472, 0.09669710478946494,
-0.4180622555915834, 0.1638895641613929, 0.01987012194056022,
-0.2253058108757411 ],
[ 0.2329417268747001, 0.4541966100017395, 0.6332975501358535,
0.9000844496702366, -0.1286729545532158, -0.7430657044481105,
0.01844125817952035, 0.4014622238287600, -0.1953120508344944,
-0.1021227144595396 ]]
```

Fock operator, transformation to the orthogonal basis set and diagonalization

In the first SCF step, the density matrix is set to zero; i.e. the problem is treated as a one-electron problem. The Fock operator in a one-electron problem is $\mathbf{F} = H_{core} = \mathbf{T} + \mathbf{V} = \mathbf{TVMatrix}$.

Transformation to an orthonormal basis set yields $\mathbf{F}' = \mathbf{A}^T \mathbf{F} \mathbf{A} = \mathbf{A}^T \mathbf{TVMatrix} \mathbf{A}$.

```
> TVMatrix := simplify(TMatrix + VMatrix);    #, shape=symmetric
TVMatrix := [ [ 13.20725102243175, -0.097170792449109, -0.8252629660074821,
-0.3967977528882390, -3.283005506175217 × 10-33, -1.197547162464358
× 10-11, -0.0002564523808213624, -2.075004113358039 × 10-12,
-0.003273494575505091, -0.1444252406912177 ],
[ -0.097170792449109, -0.244299622022847, -1.133206375671994,
-0.7328157203765911, -1.112156971617272 × 10-6, -0.0001131805703607876,
-0.008893839072440023, -0.00008493115514905797, -0.02822599610483964,
-0.2939536412014465 ],
[ -0.8252629660074821, -1.133206375671994, -1.394231724904161,
-1.161635368053535, -0.04532033785455725, -0.09542727345633053,
-0.1894538869895805, -0.09170072849243920, -0.2457795799513420,
-0.5839418080047903 ],
[ -0.3967977528882390, -0.7328157203765911, -1.161635368053535,
-1.340259028787493, -0.2873255240137540, -0.4631269817657455,
-0.6379408067379752, -0.4533130181858473, -0.7101649346209281,
-0.9713723236791492 ],
[ -3.283005506175217 × 10-33, -1.112156971617272 × 10-6,
-0.04532033785455725, -0.2873255240137540, 16.99621689540911,
-4.632689994491371, -3.940491343771723, -4.404379041402708,
```

(17.7.2.1)

$-3.158245899519811, -0.5265645821225996]$,
 $[-1.197547162464358 \times 10^{-11}, -0.0001131805703607876,$
 $-0.09542727345633053, -0.4631269817657455, -4.632689994491371,$
 $-3.745045240296371, -4.159508131806567, -3.678677578213270,$
 $-3.772224633570489, -0.8372748748820350]$,
 $[-0.0002564523808213624, -0.008893839072440023, -0.1894538869895805,$
 $-0.6379408067379752, -3.940491343771723, -4.159508131806567,$
 $-3.843498033220363, -4.173193572407878, -3.636643453253681,$
 $-1.122371331696284]$,
 $[-2.075004113358039 \times 10^{-12}, -0.00008493115514905797,$
 $-0.09170072849243920, -0.4533130181858473, -4.404379041402708,$
 $-3.678677578213270, -4.173193572407878, -3.593415120761269,$
 $-3.765039299279081, -0.8204023513607774]$,
 $[-0.003273494575505091, -0.02822599610483964, -0.2457795799513420,$
 $-0.7101649346209281, -3.158245899519811, -3.772224633570489,$
 $-3.636643453253681, -3.765039299279081, -3.479540768759335,$
 $-1.228268909483475]$,
 $[-0.1444252406912177, -0.2939536412014465, -0.5839418080047903,$
 $-0.9713723236791492, -0.5265645821225996, -0.8372748748820350,$
 $-1.122371331696284, -0.8204023513607774, -1.228268909483475,$
 $-1.400931468412453]]$

> FockOp := simplify(Transpose(TransMatrix).TVMatrix.TransMatrix);
FockOp := $[[31.11863782698329, -10.85254070020611, 0.4213386914251993,$ **(17.7.2.2)**
 $6.905924563469471, 12.84123003579721, 5.047384160445426,$
 $-7.160914644491238, 2.343767213310648, -0.5661466081205512,$
 $0.5779643453580394]$,
 $[-10.85254070020634, 8.194198037992084, -0.0153174422094001,$
 $-5.563802565724111, -9.611834511763877, -3.492873449894303,$
 $3.752572297340760, -1.028526609343675, 0.05268610611315153,$
 $-0.1768633002869870]$,
 $[0.4213386914251801, -0.01531744220939870, 3.507832078292834,$
 $-3.813910226332141, 3.382712497355533, -3.681484727402834,$

-2.225898635856068, -2.060866718857321, -0.5885825498815500,
 -0.01455825927517334],
 [6.905924563469516, -5.563802565724112, -3.813910226332140,
 9.239746083724128, 6.389012822853614, 9.315460948524701,
 -2.893449757233383, 5.168684896768130, 0.6114054946087052,
 0.4476835962145716],
 [12.84123003579745, -9.611834511763892, 3.382712497355532,
 6.389012822853618, 20.18774456124387, 3.899843608426217,
 -13.61805783581703, 1.427428224356951, -1.733118466563610,
 0.7258998526431361],
 [5.047384160445488, -3.492873449894294, -3.681484727402835,
 9.315460948524701, 3.899843608426217, 9.120174653162092,
 -1.809599096446725, 6.552159252964674, 1.150674431370055,
 0.7265931255923441],
 [-7.160914644491248, 3.752572297340765, -2.225898635856069,
 -2.893449757233386, -13.61805783581706, -1.809599096446730,
 10.47564368652773, -0.1872755110430748, 1.869735600728588,
 -0.8647878555231815],
 [2.343767213310702, -1.028526609343680, -2.060866718857319,
 5.168684896768131, 1.427428224356950, 6.552159252964674,
 -0.1872755110430704, 3.540959974907777, 1.095041149805177,
 0.1957884004823157],
 [-0.5661466081205660, 0.0526861061131524, -0.5885825498815533,
 0.6114054946087052, -1.733118466563611, 1.150674431370056,
 1.869735600728588, 1.095041149805179, -1.066222372387203,
 0.5986572301082256],
 [0.5779643453580133, -0.1768633002869870, -0.0145582592751723,
 0.447683596214572, 0.7258998526431342, 0.7265931255923444,
 -0.8647878555231802, 0.1957884004823164, 0.5986572301082254,
 -4.590599668287990]]

Fock operator **F'** is diagonalized and yields as eigenvalues the orbital energies and as eigenvectors the coefficients **c'** representing the molecular orbitals in the orthonormal basis set.

> EValues, OBEigenVec := Seig(FockOp);

$$\begin{aligned}
 EValues, OBEigenVec := & \begin{bmatrix} -4.803547503823014 \\ -1.508924623116950 \\ -1.180528447988731 \\ -0.6134088476417371 \\ -0.4533204493084998 \\ 1.739819602176126 \\ 4.151263711047467 \\ 15.81151751886472 \\ 20.80676059773421 \\ 55.77848330421527 \end{bmatrix}, \left[\begin{array}{l} 0.006473587694871442, \\ -0.01582444592951555, -0.02629036681744586, -0.004847588497328674, \\ -0.06053398742835095, -0.003289325571946853, -0.2293026272190449, \\ -0.7364341030419510, 0.01264634572532710, -0.6326476118204253 \end{array} \right], \\
 & \left[\begin{array}{l} 0.02189693493660426, -0.1064992655428865, -0.1856535543541457, \\ -0.02721315176302633, -0.4885026452104336, 0.01655270582161274, \\ -0.7804376757954305, 0.01512883402931476, 0.01884227010030887, \\ 0.3231020392937847 \end{array} \right], \\
 & \left[\begin{array}{l} 0.0001839342414997860, 0.08435862859671232, 0.05888725478530983, \\ 0.5908237240963662, -0.03661677867130644, 0.7081935206640151, \\ -0.004205623057903761, 0.01045070966160430, 0.3703798725028872, \\ -0.01249801882217417 \end{array} \right], \\
 & \left[\begin{array}{l} -0.007996436720974174, -0.04490124427329285, 0.2570425915597967, \\ 0.5088430895119107, -0.5151542678796894, -0.2338273451349944, \\ 0.1234380861338318, 0.2225288992468815, -0.4603601182745313, \\ -0.2760092280866588 \end{array} \right], \\
 & \left[\begin{array}{l} 0.002011220091785774, -0.07988533452417018, -0.2506337748389830, \\ -0.4014940781741348, -0.3838521316834898, 0.1544085474356251, \\ 0.1320713601485786, 0.4506536319997850, 0.3367399174521471, \\ -0.5142856537877077 \end{array} \right], \\
 & \left[\begin{array}{l} 0.05159830760275818, 0.06503720783715842, -0.6120869354143397, \\ 0.1282008494571304, 0.3600809431033394, 0.2280220011779348, \\ -0.1849532492507908, 0.2234057773990351, -0.5381165765009339, \\ -0.2160608424649647 \end{array} \right],
 \end{aligned}
 \tag{17.7.2.3}$$

```
[ -0.07519522364416312, 0.08279153325654587, -0.2791587133788906,
-0.2122219187172027, -0.4450132112707277, 0.3471252006158482,
0.4598100658629035, -0.3710142373959815, -0.3174753387527293,
0.3100377323474599 ],
[ -0.03902239826975188, 0.2178355313851829, 0.6054455243292844,
-0.4074502477125401, 0.05377767316632438, 0.4508393777772958,
-0.2338811650636927, 0.1250653487565193, -0.3622943949778094,
-0.1034301419256422 ],
[ 0.1924438282272437, -0.9354173919817102, 0.1138360150291453,
-0.04378600004729135, 0.09584386151500949, 0.2092533762633406,
0.05789739404484028, -0.03073743705962757, -0.1236252495853142,
0.02303696509082001 ],
[ -0.9759797710345224, -0.1983709514617763, -0.01956419223883298,
0.02525804826518406, 0.06213297902550174, 0.01126235973070877,
-0.04420789656021856, 0.02389704308651693, -0.008837802327131461,
-0.02237999467399400 ]]
```

back transformation of the coefficients and formation of the density matrix

We make a transformation of the coefficients back to the original AO-Basis: $\mathbf{c} = \mathbf{A} \mathbf{c}'$.

```
> MOEigenVectors := simplify(TransMatrix.OBEigenVec);
```

```
MOEigenVectors := [ [ 0.00018797008536893, 0.01961064825101008,
-0.01210340149449655, 0.01984172545886070, -0.0004082276214428069,
0.002989362003916612, 0.0005757988852733591, 0.01103451885010599,
1.261268763662401, -0.003715245216982453 ],
[ -0.00143981498160967, 0.1106586309686407, -0.08868025812600926,
0.08258091521560699, -0.04853839908510058, -1.479975691801737,
0.08818553315754896, 0.01448806326184052, -1.054225971239010,
0.01393169163258189 ],
[ 0.00339633958731257, 0.5621471712890172, -0.3837701996750693,
1.211264100069166, 0.05052844717507538, 1.697951010379736,
-0.1314178733946133, -0.03702291654001666, 0.6046056267222948,
-0.02111964289323165 ],
[ -0.00080050214481114, 0.2324612643026341, -0.3508836666042131,
-1.809468916531089, 0.1775302904404886, -0.8564609885307930,
```

(17.7.3.1)

```

0.09246143946899619, 0.03127101753982390, -0.2574829017632981,
0.01639263328935819 ],
[ 0.09997886565613699, -0.01584855806789260, -0.02618739765019025,
0.000548997582079992, -0.03528121702413485, 0.007688540623877570,
0.1587333692914303, -0.02917234815899937, 0.000585586680496511,
1.744369410887052 ],
[ -0.1302410538849142, 0.8201919942446164, 1.353045128014425,
0.2697496293983348, 3.142369728745981, 0.2394255139786812,
13.89677534668563, 44.13389958565267, -0.7518851133784502,
37.19226124400664 ],
[ 0.4953918506852108, -0.8659090868987790, -1.497451614911085,
-0.2397286613534199, -3.534838497781301, 0.01918448531234994,
-7.967815722463424, -7.110915821504973, 0.2369574586011290,
-3.854259531231146 ],
[ 0.5740817442033509, -0.8213811170693604, -1.356461116480445,
-0.2385016765876476, -3.010775363767098, -0.1651738193431315,
-11.21961688059469, -41.22805511364732, 0.6814292991601997,
-36.67985643323016 ],
[ 0.0883845638661395, 0.8289627965042364, 1.469142582601103,
0.2832998198384439, 3.920021892308649, -0.09846302965712052,
5.585586707857475, 4.064350554407508, -0.1723342814422970,
2.108616569212632 ],
[ 0.01080316804746511, 0.2148395714508902, 0.8829563250145110,
0.6135624083474154, -0.9527027595625479, 0.2008578489729770,
-0.2712150061277586, -0.1348236569351580, 0.06197417510184064,
-0.06826239848153544 ]]

```

This transformation takes the occupation matrix into the MO basis (input) into the natural AO basis --> density matrix

```

> DensityMatrix := MOEigenVectors.Occupation.Transpose
(MOEigenVectors);

```

```

DensityMatrix := [ [ 0.0007692257151556766, 0.004339633691438670,
0.02204941770338280, 0.009117131231535091, -0.0005840149232853537,
0.03212003055081674, -0.03377583934412181, -0.03199981194480462,

```

(17.7.3.2)

0.03254622293886689, 0.008430347877082584],
[0.004339633691438670, 0.02449481135007015, 0.1244030925542323,
0.05144999567189949, -0.003795461614472094, 0.1818976924703053,
-0.1930671734158170, -0.1834389628287216, 0.1832092615317944,
0.04751659658288278],
[0.02204941770338279, 0.1244030925542323, 0.6320419546216718,
0.2613494467697403, -0.01713931981510602, 0.9212525332626123,
-0.9701716496797800, -0.9195745899131251, 0.9325985503037109,
0.2416162971987412],
[0.009117131231535091, 0.05144999567189949, 0.2613494467697404,
0.1080777604097259, -0.007528418284459096, 0.3815342523919593,
-0.4033737667012501, -0.3827976932315703, 0.3852619754045579,
0.09986646088503456],
[-0.0005840149232853537, -0.003795461614472094, -0.01713931981510602,
-0.007528418284459096, 0.02049390074143851, -0.05204042655375994,
0.1265042514640659, 0.1408274958181838, -0.00860255315934201,
-0.004649617873084817],
[0.03212003055081675, 0.1818976924703053, 0.9212525332626128,
0.3815342523919594, -0.05204042655375995, 1.379355279080028,
-1.549464115074569, -1.496918455650247, 1.336794700948639,
0.3496053611183334],
[-0.03377583934412181, -0.1930671734158170, -0.9701716496797804,
-0.4033737667012501, 0.1265042514640658, -1.549464115074569,
1.990423264998390, 1.991273581565842, -1.348042851056779,
-0.3613594714649364],
[-0.03199981194480462, -0.1834389628287217, -0.9195745899131256,
-0.3827976932315703, 0.1408274958181838, -1.496918455650247,
1.991273581565842, 2.008473577011344, -1.260308846433329,
-0.3405265312668484],
[0.03254622293886689, 0.1832092615317944, 0.9325985503037114,
0.3852619754045579, -0.00860255315934202, 1.336794700948639,
-1.348042851056780, -1.260308846433329, 1.389982298235863,

```
0.3580976904918989 ],
[ 0.008430347877082584, 0.04751659658288278, 0.2416162971987412,
0.09986646088503456, -0.004649617873084817, 0.3496053611183334,
-0.3613594714649364, -0.3405265312668483, 0.3580976904918989,
0.09254549980212785 ]]
```

further SCF steps

two-electron operator for closed shells

M_{mn} = element mn of Matrix \mathbf{M} = element mn of Matrix $(2\mathbf{J} - \mathbf{K}) = S_l S_r P_{lr} [(mn|lr) - \frac{1}{2} (ml|nr)]$

```
> MMatrix := Matrix(Dim, datatype=float[8]):
makeMMatrix := proc(MMatrix,Dim)
    local ro,colm;
    for ro to Dim do
        for colm to ro do
            MMatrix[ro,colm] := MOperator(ro,colm);
            MMatrix[colm,ro] := MMatrix[ro,colm];
        end do;
    end do;
end proc;
evalhf(makeMMatrix(MMatrix,Dim)):
```

```
> MMatrix;
```

```
[ [ 2.33824637256615064, 1.13094516313224736, 0.311747512127070359,
0.0640243514993241586, 0.00679339303772186643, 0.00764693805456988836,
-0.00618381987964088803, 0.00798901582878459686, -0.0144026992055467911,
-0.000279859135581706159 ],
[ 1.13094516313224736, 1.83607538535892378, 1.01926158091762664,
0.351356510985358805, 0.0178966373142476209, 0.0199023713532779471,
-0.0154547023557731703, 0.0208275238239824627, -0.0315085254339516480,
0.0757631926422286522 ],
[ 0.311747512127070359, 1.01926158091762664, 1.37465813777334778,
0.908375830321041966, 0.0210745580678120444, 0.0434384396354185395,
0.0621081898097517321, 0.0425740884242332696, 0.0840967623908374512,
0.356010117600178810 ],
[ 0.0640243514993241586, 0.351356510985358805, 0.908375830321041966,
1.16425623472518103, 0.0232425467819018620, 0.143480294546840109,
0.316253084566058607, 0.134651040351289469, 0.392736071416025956,
```

(17.8.1.1)

```

0.758595044447497702 ],
[ 0.00679339303772186643, 0.0178966373142476209, 0.0210745580678120444,
0.0232425467819018620, 4.97601631958311419, 1.96789785284174190,
0.527905203119399036, 2.11068194550710109, 0.305850629912793825,
0.0186752241979959155 ],
[ 0.00764693805456988836, 0.0199023713532779471, 0.0434384396354185395,
0.143480294546840109, 1.96789785284174190, 3.01241496033975320,
1.92716593525196100, 3.04026391528691642, 1.49552511193637927,
0.233936433407707312 ],
[ -0.00618381987964088803, -0.0154547023557731703, 0.0621081898097517321,
0.316253084566058607, 0.527905203119399036, 1.92716593525196100,
2.41829595464512126, 1.85927073822422995, 2.28186092607314395,
0.576865851765403059 ],
[ 0.00798901582878459686, 0.0208275238239824627, 0.0425740884242332696,
0.134651040351289469, 2.11068194550710109, 3.04026391528691642,
1.85927073822422995, 3.07823675738157654, 1.42523785547717341,
0.216924024792927855 ],
[ -0.0144026992055467911, -0.0315085254339516480, 0.0840967623908374512,
0.392736071416025956, 0.305850629912793825, 1.49552511193637927,
2.28186092607314395, 1.42523785547717341, 2.30081437916611220,
0.722938661018480810 ],
[ -0.000279859135581706159, 0.0757631926422286522, 0.356010117600178810,
0.758595044447497702, 0.0186752241979959155, 0.233936433407707312,
0.576865851765403059, 0.216924024792927855, 0.722938661018480810,
1.18646506506044735 ]]

```

Fock operator

In the first SCF step, the density matrix is set to zero; i.e. the problem is treated as a one-electron problem. The Fock operator in a one-electron problem is $\mathbf{F} = H_{core} = \mathbf{T} + \mathbf{V} = \mathbf{TVMatrix}$.

Transformation to the orthonormal basis set yields $\mathbf{F}' = \mathbf{A}^T \mathbf{F} \mathbf{A} = \mathbf{A}^T \mathbf{TVMatrix} \mathbf{A}$. $\mathbf{F} = H_{core} + \mathbf{M}$.

```
> FMatrix := TVMatrix + MMatrix;
```

```
FMatrix := [[ 15.54549739499790, 1.033774370683138, -0.5135154538804117,
-0.3327734013889148, 0.006793393037721866, 0.007646938042594416,
-0.006440272260462250, 0.007989015826709593, -0.01767619378105188,
```

(17.8.2.1)

—0.1447050998267994],
[1.033774370683138, 1.591775763336077, —0.113944794754367,
—0.3814592093912323, 0.01789552515727600, 0.01978919078291716,
—0.02434854142821319, 0.02074259266883340, —0.05973452153879129,
—0.2181904485592178],
[—0.5135154538804117, —0.113944794754367, —0.019573587130813,
—0.2532595377324930, —0.02424577978674521, —0.05198883382091199,
—0.1273456971798288, —0.04912664006820593, —0.1616828175605046,
—0.2279316904046115],
[—0.3327734013889148, —0.3814592093912323, —0.2532595377324930,
—0.176002794062312, —0.2640829772318521, —0.3196466872189054,
—0.3216877221719166, —0.3186619778345578, —0.3174288632049021,
—0.2127772792316515],
[0.006793393037721866, 0.01789552515727600, —0.02424577978674521,
—0.2640829772318521, 21.97223321499222, —2.664792141649629,
—3.412586140652324, —2.293697095895607, —2.852395269607017,
—0.5078893579246037],
[0.007646938042594416, 0.01978919078291716, —0.05198883382091199,
—0.3196466872189054, —2.664792141649629, —0.732630279956618,
—2.232342196554606, —0.638413662926354, —2.276699521634110,
—0.6033384414743277],
[—0.006440272260462250, —0.02434854142821319, —0.1273456971798288,
—0.3216877221719166, —3.412586140652324, —2.232342196554606,
—1.425202078575242, —2.313922834183648, —1.354782527180537,
—0.5455054799308809],
[0.007989015826709593, 0.02074259266883340, —0.04912664006820593,
—0.3186619778345578, —2.293697095895607, —0.638413662926354,
—2.313922834183648, —0.515178363379692, —2.339801443801908,
—0.6034783265678495],
[—0.01767619378105188, —0.05973452153879129, —0.1616828175605046,
—0.3174288632049021, —2.852395269607017, —2.276699521634110,
—1.354782527180537, —2.339801443801908, —1.178726389593223,

$-0.5053302484649942]$,
 $[-0.1447050998267994, -0.2181904485592178, -0.2279316904046115,$
 $-0.2127772792316515, -0.5078893579246037, -0.6033384414743277,$
 $-0.5455054799308809, -0.6034783265678495, -0.5053302484649942,$
 $-0.214466403352006]]$

$\mathbf{F}' = \mathbf{A}^T \mathbf{F} \mathbf{A}$, transformation of the Fock operator to the orthonormal basis set

> FockOp := Transpose(TransMatrix).FMatrix.TransMatrix;
FockOp := $[[34.57857973431783, -11.65248042044669, 0.4225720743840210,$
 $7.140279149413125, 13.29783774487541, 5.194155598150843,$
 $-7.222172752366065, 2.293474824023864, -0.5214029262639450,$
 $0.5465434032167849]$,
 $[-11.65248042044550, 10.68313572777568, -0.06606050253784364,$
 $-6.019362113674717, -10.21816460573978, -3.426689828976289,$
 $3.611529794993229, -1.034638535083967, 0.06668229350436662,$
 $-0.1288097633365712]$,
 $[0.4225720743840345, -0.06606050253784148, 4.965636534831982,$
 $-4.125543519652312, 3.641244602684468, -3.736722591763006,$
 $-2.174812834470610, -1.990206695929177, -0.6020773356713837,$
 $-0.02568116683523768]$,
 $[7.140279149413057, -6.019362113674705, -4.125543519652318,$
 $11.31151587785336, 7.294170649796723, 10.04552931299159,$
 $-3.175376747235731, 5.201970377401596, 0.5964133444026851,$
 $0.4357903938023041]$,
 $[13.29783774487546, -10.21816460573978, 3.641244602684469,$
 $7.294170649796721, 23.66872344631034, 4.508179850016177,$
 $-14.60856789892395, 1.553400855949703, -1.686234462753812,$
 $0.7092841526528845]$,
 $[5.194155598150831, -3.426689828976295, -3.736722591763013,$
 $10.04552931299160, 4.508179850016172, 11.25312383694304,$
 $-2.483537265303887, 7.231982345217457, 1.077259363902363,$
 $0.8021296325702990]$,
 $[-7.222172752366084, 3.611529794993220, -2.174812834470608,$
 $-3.175376747235732, -14.60856789892396, -2.483537265303895,$

(17.8.2.2)

14.17428290013402, -0.9723450730876998, 2.383590855135251,
 -1.311761156161448],
 [2.293474824023863, -1.034638535083963, -1.990206695929178,
 5.201970377401597, 1.553400855949703, 7.231982345217453,
 -0.9723450730876985, 5.846091304706129, 1.180511817012493,
 0.6107435212680582],
 [-0.5214029262639499, 0.06668229350436607, -0.6020773356713832,
 0.5964133444026866, -1.686234462753811, 1.077259363902360,
 2.383590855135249, 1.180511817012492, 0.4546915257410570, 0.2135541065890498
],
 [0.5465434032168449, -0.1288097633365603, -0.0256811668352378,
 0.4357903938023031, 0.7092841526528824, 0.8021296325702979,
 -1.311761156161447, 0.6107435212680585, 0.2135541065890500,
 -1.987401047214157]]

Fock operator \mathbf{F} is diagonalized and yields as eigenvalues the orbital energies and as eigenvectors the coefficients \mathbf{c} representing the molecular orbitals in the orthonormal basis set.

> EValues, OBEigenVec := Seig(FockOp);

EValues, OBEigenVec :=
$$\begin{bmatrix} -2.207501583464950 \\ -0.2268876099434188 \\ 0.09130674914273735 \\ 0.6917827597715717 \\ 1.683498093978440 \\ 3.710687904359999 \\ 7.209896102601778 \\ 19.76617681456039 \\ 23.17928182615314 \\ 61.05013878423960 \end{bmatrix}, \left[\begin{array}{l} 0.002323401112191624, \\ 0.002334755172362391, 0.004320290219692166, 0.008392559903417499, \\ -0.09295811141576121, 0.003386661863714327, 0.2644136229417373, \\ 0.7329474547982843, 0.01573917195658887, -0.6195749460503869], \\ -0.004595616893782023, -0.01057072978060088, 0.03845446843628092, \\ 0.05146249391795587, -0.6232510219267010, 0.04297183964475622, \\ 0.7051478360968391, -0.06217652408422750, 0.01756552997789946, \\ 0.3225579815730988], \end{array} \right. \quad (17.8.2.3)$$

[0.005302761176285347, -0.005176351990946333, 0.2307054652367822,
-0.6170483979925453, -0.001658827779198035, 0.6577902038043560,
-0.01350775830206812, -0.01104139252743196, 0.3644470679061504,
-0.01243413967372800],
[-0.02335072729042336, -0.03056981628795611, 0.5729099729506348,
-0.3354725833688752, -0.3464404030287892, -0.2739346932773680,
-0.1757352824726286, -0.2050556342467362, -0.4601618656608331,
-0.2793057899263922],
[-0.04503693639385572, 0.06077646425274561, -0.1671010246510426,
0.3582720021593371, -0.4341767506735564, 0.1874173203662799,
-0.2192225204394146, -0.4273486710459263, 0.3347935161048502,
-0.5211425087683792],
[0.04104229165844642, -0.02976071747350678, -0.6467793434335756,
-0.3361646160331093, 0.03532313474375723, 0.2063038832695908,
0.1727220789281601, -0.2262275179970654, -0.5409293251269198,
-0.2206225385017361],
[-0.1345121267016493, 0.2061810972143838, -0.1160063289193317,
0.1332799274989482, -0.4224697569714048, 0.3457727503647862,
-0.4979230501726435, 0.4003313401090916, -0.3175059198622391,
0.3180138206643924],
[0.02274548463059347, 0.2724826376352205, 0.3890375141690465,
0.4476388102242379, 0.3241792085729628, 0.4896210878064988,
0.2524597545034300, -0.1410892148799477, -0.3656265002864839,
-0.1065803099054649],
[0.1505826794615591, -0.9217129976000151, 0.07729268778914408,
0.2108992028462049, -0.02664548029591599, 0.2309811837967038,
-0.07448329981112588, 0.03909985309778999, -0.1249122829130923,
0.02398844681560112],
[-0.9769346438593159, -0.1674208433403509, 0.004863461282573401,
-0.001613425687681814, 0.09409164420256958, 0.009344897209691447,
0.08175654533857895, -0.03530523172507862, -0.009277080299578948,
-0.02419680805876691]]

calculation of density matrix

We transform the coefficients back to the original atomic-orbital basis: $\mathbf{c} = \mathbf{A} \mathbf{c}'$

```
> MOEigenVectors := simplify(TransMatrix.OBEigenVec);
```

(17.8.3.1)

```
MOEigenVectors := [[ 0.00036670287251517, 0.01528457621042190,  
  -0.004971726919448222, -0.02799272959438948, -0.002868971447830323,  
  0.009436208126825013, -0.000411374446239972, -0.01627160660003979,  
  1.261131393351838, -0.003468303497257089 ],  
 [ -1.24181912019  $\times 10^{-6}$ , 0.09476425610087931, -0.05144522214177622,  
  0.01609612875316152, -0.06136211544285489, -1.493419856769101,  
  -0.05767958852054606, -0.007522746631184137, -1.043875165890855,  
  0.01323940072783405 ],  
 [ 0.01023083624869643, 0.3417873611217311, -0.02861016124699695,  
  -1.491281110127392, -0.05873709986045800, 1.580192317663088,  
  0.09002240850921068, 0.03017298480567873, 0.5908870924769543,  
  -0.02015962303121152 ],  
 [ -0.00495743997330009, 0.4849097634705529, -0.8891144012654938,  
  1.636723962451246, 0.08657026475826232, -0.7166574040709792,  
  -0.06384142064304792, -0.02632884200205577, -0.2486272352179861,  
  0.01559130096834038 ],  
 [ 0.08800876359278427, -0.01229400292713231, -0.01160017024640991,  
  -0.001329496263761585, -0.03076906599487457, 0.003732842107457957,  
  -0.1637708059240356, 0.009091836353025991, 0.000135383668111348,  
  1.745053569309756 ],  
 [ 0.0610397261446381, 0.1004968358685007, -0.2622074177957731,  
  -0.4669535056698550, 5.026090044065975, -0.1589256716519914,  
  -16.07792350617987, -43.89293119728695, -0.9369322801991040,  
  36.41046186635824 ],  
 [ 0.3102112764571839, -0.1659387037690532, 0.1561795926791595,  
  0.4093742582417198, -4.880811742765490, 0.2748701244259699,  
  7.829021102618213, 6.727574566379529, 0.2584383100561738,  
  -3.730807742070419 ],  
 [ 0.3298467343334321, -0.1474834237435292, 0.1802667757852738,
```



```

0.4104458797588639, -4.631493803847888, 0.1609046323094461,
13.24127702962508, 41.15250038260570, 0.8570682396026414,
-35.94578270306507 ],
[ 0.3393708512063385, 0.02487563276289599, -0.3274829897564361,
-0.4762484803770753, 4.863152499513422, -0.2943658765469411,
-5.262078337201515, -3.800799590440006, -0.1828453645195160,
2.036627784110049 ],
[ 0.03309990461881089, 0.2884680937601078, 1.320797459415102,
-0.3225658131353624, -0.4700389586864577, 0.1697793653097644,
0.2133708661996542, 0.1207333482207004, 0.05983440709537875,
-0.06523701845852243 ]]

```

We transform the occupation matrix from the MO basis (see input) to the original atomic-orbital basis
-> density matrix **P**.

```

> DensityMatrix := MOEigenVectors.Occupation.Transpose
(MOEigenVectors);

```

```

DensityMatrix := [ [ 0.0004675054818578120, 0.002896862078038380,
0.01045565329172944, 0.01481964465492959, -0.0003112711165098424,
0.003116869979306276, -0.004845094795706541, -0.004266531770024203,
0.001009323541464469, 0.008842500786910388 ],
[ 0.002896862078038379, 0.01796052847179031, 0.06477842463307075,
0.09190423833516805, -0.002330282665714200, 0.01904686418254323,
-0.03145088609462582, -0.02795313309649546, 0.004713798793205273,
0.05467284641984178 ],
[ 0.01045565329172944, 0.06477842463307075, 0.2338465404658085,
0.3313706193643670, -0.006603063138638756, 0.06994607155085931,
-0.1070840617945056, -0.09406672456873256, 0.02394844896907253,
0.1978667764761717 ],
[ 0.01481964465492959, 0.09190423833516806, 0.3313706193643670,
0.4703241098403128, -0.01279556042827520, 0.09685859226436986,
-0.1640063027546781, -0.1463026950182982, 0.02076005315138610,
0.2794338086474909 ],
[ -0.0003112711165098424, -0.002330282665714200, -0.006603063138638756,
-0.01279556042827520, 0.01579336995420586, 0.008273044867393724,

```

(17.8.3.2)

0.05868272360679483, 0.06168512981401877, 0.05912357582419859,
 —0.001266691817060816],
 [0.003116869979306276, 0.01904686418254323, 0.06994607155085931,
 0.09685859226436986, 0.008273044867393734, 0.02765092437478557,
 0.00451779337002965, 0.01062427380827580, 0.04643005240401096,
 0.06202107957050937],
 [—0.004845094795706541, —0.03145088609462582, —0.1070840617945056,
 —0.1640063027546781, 0.05868272360679485, 0.00451779337002965,
 0.2475333788994980, 0.2535907693124617, 0.2022976693779387,
 —0.07520011578973925],
 [—0.004266531770024203, —0.02795313309649546, —0.09406672456873256,
 —0.1463026950182982, 0.06168512981401878, 0.01062427380827580,
 0.2535907693124617, 0.2611004568590863, 0.2165432470214185,
 —0.06325273332649446],
 [0.001009323541464469, 0.004713798793205273, 0.02394844896907253,
 0.02076005315138610, 0.05912357582419860, 0.04643005240401096,
 0.2022976693779387, 0.2165432470214185, 0.2315827435077384,
 0.03681793833904711],
 [0.008842500786910388, 0.05467284641984178, 0.1978667764761717,
 0.2794338086474909, —0.001266691817060814, 0.06202107957050937,
 —0.07520011578973925, —0.06325273332649446, 0.03681793833904711,
 0.1686188896067295]]

convergence

energy = trace of '**EMatrix**' = trace $[\mathbf{P} (\mathbf{H} + \frac{1}{2} \mathbf{M})]$

```
> EMatrix := 1/2*DensityMatrix.(TVMatrix + FMatrix);
EMatrix := [ [ —0.005606104417435970, —0.01485017528803681,
  —0.02294075848505203, —0.02333306443957430, 0.002635495525251565,
  0.002889079556929459, —0.000924110776963050, 0.002993779334530468,
  —0.003268428396761108, —0.01685109985514146 ],
  [ —0.03472460410381030, —0.09194718097917035, —0.1414751564615882,
  —0.1422904215387038, 0.02447988878399065, 0.03251760097900530,
  0.00919112578423677, 0.03297709790486759, —0.00639339300189419,
```

(17.8.4.1)

—0.1003584532586696],
[—0.1253910236351214, —0.3321839810799389, —0.5136678652612325,
—0.5238840180569139, 0.0516671858105538, 0.0515657896010317,
—0.0339846489952389, 0.0540763407007798, —0.08547014994798505,
—0.3804937865214396],
[—0.1776139285086598, —0.4702253241074635, —0.7222815498908537,
—0.7229338140792472, 0.1429801855188404, 0.1981813672584047,
0.0795057280382011, 0.2001198655900062, —0.0025252439800050,
—0.5045720068960704],
[0.003220426528563053, 0.007139175640768434, —0.01094768203604781,
—0.07340242538115043, —0.3180943528214488, —0.5692776917549216,
—0.5784490565207395, —0.5620003711281096, —0.5358132544145559,
—0.1463089062960143],
[—0.03771609377782000, —0.1008311543270999, —0.1703564418305231,
—0.2146117691100685, —0.1924605629634140, —0.3574292369063500,
—0.3906267951775631, —0.3518531910472236, —0.3789850055091064,
—0.2169366801937849],
[0.05627766815947909, 0.1441244692044332, 0.1444515930753178,
—0.07463153323342541, —1.152877793225421, —2.048637628042585,
—2.050784852718986, —2.023581321432579, —1.880340793725331,
—0.3858605750118669],
[0.04922849064421489, 0.1251492529861870, 0.1103618793783725,
—0.1228379660021370, —1.218372179491846, —2.168375826520369,
—2.177787280729875, —2.141593081830836, —2.001316864852899,
—0.4409533202198281],
[—0.01406832611326618, —0.04260447106203195, —0.1501275773905532,
—0.3915820227739177, —1.207935626575373, —2.170396266716141,
—2.223554673410346, —2.141983603560972, —2.071033473557020,
—0.6407335316369760],
[—0.1061869301441655, —0.2816942250113850, —0.4416804323842532,
—0.4676937501599412, —0.04393565890753295, —0.1135566266808108,
—0.1891518303400464, —0.1093981659323534, —0.2213154392508146,

−0.3654272814711249]]

We save the energy in `Etot_old`, for a comparison in the next step.

```
> Eelec_old := Eelec;  
Eelec_old := Eelec
```

(17.8.4.2)

We form a trace.

```
> Eelec := Trace(EMatrix);  
Eelec := −8.638517244042852
```

(17.8.4.3)

We test the energy decrease in the last SCF step toward convergence.

```
> Dif := Eelec - Eelec_old;  
Dif := 0.
```

(17.8.4.4)

For this diatomic molecule, we calculate the contribution of nuclear repulsion to the total energy.

```
> Enucl := AtomicNumber[1]*AtomicNumber[2]/aa;  
Enucl := 1
```

(17.8.4.5)

```
> Etotal := Eelec + Enucl;  
Etotal := −7.638517244042852
```

(17.8.4.6)

To undertake the next iteration, click on **next SCF calculation** below and execute the statements beginning with that indicated by the cursor; repeat this iterative process until a satisfactory convergence is achieved such that **Dif** is acceptably small.

next SCF calculation

next SCF step

We transform H_{core} , i.e. **TVMatrix**, to a representation in the molecular orbitals for the calculation of configuration interaction.

```
> Hij := Transpose(MOEigenVectors).TVMatrix.MOEigenVectors;  
Hij := [[ −4.751823774375723, 0.1225465017063171, 0.104407991736267,  
−0.0077454089701798, 0.3340732258026806, −0.0249221342822876,  
0.2909536070073448, −0.1583428599151557, −0.00360589764429490,  
−0.06438877568204255 ],  
[ 0.1225465017063169, −1.452304276258762, 0.037708898754747,  
0.1591247582712391, −0.1804060077747876, 0.0316871476307017,  
−0.0455130551659810, 0.0503490934907639, −0.00727999864504102,  
0.00089080467852302 ],  
[ 0.1044079917362667, 0.0377088987547467, −1.019241646993008,  
−0.1194617768876003, −0.2761246497179818, 0.0135059507607699,  
−0.0148128822302578, 0.05485349346892059, 0.01170819110342614,  
−0.00824222494518999 ],
```

(17.8.5.1)

```
[ -0.007745408970180278, 0.1591247582712410, -0.1194617768876002,
-0.6563873665202074, -0.04853977460747906, -0.2086514924177390,
0.01316642148865705, 0.00164230868066984, -0.04216189350994592,
-0.004588141146880996 ],
[ 0.3340732258026818, -0.1804060077747900, -0.2761246497179760,
-0.04853977460747218, -0.5737918606484087, -0.00183847757159634,
-0.5993521374841023, 0.1301537634661182, -0.00150261287977077,
0.08784169637492648 ],
[ -0.02492213428228676, 0.03168714763069817, 0.01350595076077050,
-0.2086514924177386, -0.00183847757160004, 1.723906209748134,
0.06624178817664846, -0.01461695083041480, 0.1309397693080574,
-0.008831848743132963 ],
[ 0.2909536070073531, -0.04551305516598335, -0.0148128822302684,
0.01316642148865732, -0.5993521374843064, 0.06624178817663945,
4.111239718968566, 0.7427845080168158, 0.01707150886754511,
0.04658991969052544 ],
[ -0.1583428599151999, 0.05034909349077736, 0.05485349346894005,
0.00164230868064907, 0.1301537634656463, -0.01461695083044289,
0.7427845080159932, 15.77557711256807, -0.02174434336796225,
-0.7206594648298474 ],
[ -0.003605897644294116, -0.007279998645042002, 0.01170819110342540,
-0.04216189350994610, -0.00150261287977617, 0.1309397693080561,
0.01707150886753753, -0.02174434336791957, 20.80566108705726,
-0.01527389129785789 ],
[ -0.06438877568211501, 0.00089080467853969, -0.0082422249451991,
-0.00458814114685750, 0.08784169637548049, -0.00883184874309818,
0.04658991968925693, -0.7206594648364458, -0.01527389129795354,
55.76527965861307 ]]
```

function to calculate the number of occupied MO from the occupation matrix

```
> OccupiedOrbitals := proc(OccupationMatrix, Dim)
  local counter,i;
  counter := 0;
  for i to Dim do
```

```

        counter := counter + OccupationMatrix[i,i];
    end do;
    counter/2;
end proc:

```

integral functions and transformation

We transform the two-electron integrals from an atomic-orbital basis ($\mu \nu \lambda \sigma$) to a molecular-orbital basis (ijkl).

```

> IJKLIntegral := proc(i,j,k,l)
    local sumd,mu,nu,lambda,sigma;
    global Dim;
    sumd := 0;
    for mu to Dim do
        for nu to Dim do
            for lambda to Dim do
                for sigma to Dim do
                    sumd := sumd + MOEigenVectors[mu,i]*MOEigenVectors[nu,j]
                        *MOEigenVectors[lambda,k]*MOEigenVectors[sigma,l]
                        *TwoElectronIntegral(mu,nu,lambda,sigma);
                end do;
            end do;
        end do;
    end do;
    sumd;
end proc:

```

MP2 energy

We calculate the energy according to Moeller-Plesset perturbation theory of second order.

```

> MP2Energy := proc(occupied, Dim)
    local sumd,iajb,ibja,i,j,a,b;
    sumd := 0;
    for i to occupied do
        for j to occupied do
            for a from occupied+1 to Dim do
                for b from occupied+1 to Dim do
                    iajb := IJKLIntegral(i,a,j,b);
                    ibja := IJKLIntegral(i,b,j,a);
                    sumd := sumd + iajb*(2*iajb-ibja)/(EValues[i]
                        + EValues[j] - EValues[a] - EValues[b]);
                end do;
            end do;
        end do;
    end do;
    sumd;
end proc:

```

end proc:

```
> evalhf(MP2Energy(OccupiedOrbitals(Occupation,Dim),Dim));  
      -0.0265511004549089276 (17.11.1)
```

construction of configuration-interaction matrix

```
> Element11 := 2*Hij[1,1] + evalhf(IJKLIntegral(1,1,1,1));  
      Element11 := -7.850995170097753 (17.12.1)
```

```
> Element22 := 2*Hij[2,2] + evalhf(IJKLIntegral(2,2,2,2));  
      Element22 := -2.421939663406193 (17.12.2)
```

```
> Element33 := 2*Hij[3,3] + evalhf(IJKLIntegral(3,3,3,3));  
      Element33 := -1.781666785338556 (17.12.3)
```

```
> Element44 := Hij[2,2] + Hij[3,3] + evalhf(IJKLIntegral(2,2,3,3));  
      Element44 := -2.220210710964783 (17.12.4)
```

```
> Element55 := Element44;  
      Element55 := -2.220210710964783 (17.12.5)
```

```
> Element12 := evalhf(IJKLIntegral(2,1,2,1));  
      Element12 := 0.0125732760044606501 (17.12.6)
```

```
> Element13 := evalhf(IJKLIntegral(3,1,3,1));  
      Element13 := 0.00874623177470261123 (17.12.7)
```

```
> Element14 := evalhf(IJKLIntegral(3,1,2,1));  
      Element14 := 0.00939184013027870995 (17.12.8)
```

```
> Element15 := Element14;  
      Element15 := 0.00939184013027870995 (17.12.9)
```

```
> Element23 := evalhf(IJKLIntegral(3,2,3,2));  
      Element23 := 0.0472012375227964312 (17.12.10)
```

```
> Element24 := Hij[2,3] + evalhf(IJKLIntegral(3,2,2,2));  
      Element24 := -0.05276846129414777 (17.12.11)
```

```
> Element25 := Element24;  
      Element25 := -0.05276846129414777 (17.12.12)
```

```
> Element34 := Hij[2,3] + evalhf(IJKLIntegral(3,3,3,2));  
      Element34 := 0.04423424513763890 (17.12.13)
```

```
> Element35 := Element34;  
      Element35 := 0.04423424513763890 (17.12.14)
```

```
> Element45 := Element23;  
      Element45 := 0.0472012375227964312 (17.12.15)
```

```
> CIMatrix := Matrix(
```

```

[[Element11,Element12,Element13,Element14,Element15],
 [Element12,Element22,Element23,Element24,Element25],
 [Element13,Element23,Element33,Element34,Element35],
 [Element14,Element24,Element34,Element44,Element45],
 [Element15,Element25,Element35,Element45,Element55]]);
CIMatrix := [[-7.850995170097753, 0.0125732760044606501,
0.00874623177470261123, 0.00939184013027870995, 0.00939184013027870995 ],
[0.0125732760044606501, -2.421939663406193, 0.0472012375227964312,
-0.05276846129414777, -0.05276846129414777 ],
[0.00874623177470261123, 0.0472012375227964312, -1.781666785338556,
0.04423424513763890, 0.04423424513763890 ],
[0.00939184013027870995, -0.05276846129414777, 0.04423424513763890,
-2.220210710964783, 0.0472012375227964312 ],
[0.00939184013027870995, -0.05276846129414777, 0.04423424513763890,
0.0472012375227964312, -2.220210710964783 ]]

```

(17.12.16)

```
> CIEigenValues,CIEigenVec := Seig(CIMatrix);
```

```

CIEigenValues, CIEigenVec := 
$$\begin{bmatrix} -7.851068047417392 \\ -2.448421954434199 \\ -2.267411948487581 \\ -2.158172653045709 \\ -1.769948437387189 \end{bmatrix}, \begin{bmatrix} -0.9999935215446967, \\ -0.002759862099443929, 5.749559940932167 \times 10^{-18}, -0.001387251770937961, \\ 0.001848124008662634 ], \\ [0.002336081000249369, -0.9553669748073345, 4.399395917806564 \times 10^{-15}, \\ 0.2902127631127593, 0.05518186567611759 ], \\ [0.001398592874023310, 0.09394607138840865, -3.439285694762439 \times 10^{-16}, \\ 0.1213650359121092, 0.9881511562846598 ], \\ [0.001664862181731258, -0.1980409834849310, -0.7071067811865485, \\ -0.6712097867720026, 0.1012641063542285 ], \\ [0.001664862181731258, -0.1980409834849311, 0.7071067811865464, \\ -0.6712097867720025, 0.1012641063542286 ]]$$

```

(17.12.17)

```
> #DMatrix := TransMatrix.CIEigenVec;
```

calculation of electric dipolar moment

The net molecular electric dipolar moment is a sum of electronic and nuclear contributions.


```
> muex := -Trace(DensityMatrix.DMatrix) + AtomicNumber[2]*aa;
muex := 2.174720198290390 (17.13.1)
```

```
> time();
126.843 (17.1)
```

Time for convergence to $0.2 \cdot 10^{-7}$ with Maple 11 β_6 is 277.845 s, with Maple 9.5 is 175.38 s, with Maple 10.05 is 271.6 s.

3.84 density-functional theory for atoms

Implementation of density-functional theory for atoms utilizing basis sets comprising only s functions (spherical gaussians)

copyright P. Vogt, H. Huber, 2000 January

modified by J. F. Ogilvie and S. Paniagua, 2005 March -- June, 2012 March

In this program we retain the names and structure as nearly the same as possible those of the corresponding *Maple* worksheet for SCF calculations, to show the analogies and differences.

All properties have atomic units!

All properties have atomic units!

For energy 1 a.u. = $\frac{e^2}{a_0} = 1 \text{ hartree} = 2 \text{ Rydberg} = 2 * 10973731.568539 \text{ m}^{-1} = 4.35974434 \cdot 10^{-18} \text{ J}$

= $2 * 13.605691775 = 27.21138505 \text{ eV}$;

for length 1 a. u. = $a_0 = 0.52917721092 \cdot 10^{-10} \text{ m}$;

for charge 1 a. u. = $|e| = 1.602176565 \cdot 10^{-19} \text{ C}$;

for mass 1 a.u. = $m_e = 9.1093826 \cdot 10^{-31} \text{ kg} = 5.4857990945 \cdot 10^{-4} \text{ u}$,

for action or angular momentum 1 a. u. = $\frac{h}{2 \pi} = 1.05457168 \cdot 10^{-34} \text{ J s}$, and

for electric dipolar moment 1 a. u. = $|e| a_0 = 8.47835303214 \cdot 10^{-30} \text{ C m}$.

synopsis of procedures

These procedures use spherical gaussian functions of type $c e^{-\alpha r^2}$, in which α is the exponent that is fixed for each base function in the step "Exponent := ". The product of two gaussian functions is another gaussian function with an exponent that is a sum of the exponents of the original gaussians. We seek to evaluate the electronic energy by means of a self-consistent field; in matrix form, the equation to solve is

$$\mathbf{H} \mathbf{C} = \mathbf{S} \mathbf{C} \mathbf{E}$$

then

$$\mathbf{C}^{-1} \cdot \mathbf{S}^{-1} \cdot \mathbf{H} \cdot \mathbf{C} = \mathbf{E}$$

\mathbf{H} denotes the electronic hamiltonian that is applied to $\Phi(r; R)$,

$$\mathbf{H} = -\frac{1}{2m_e} \sum_{j=1}^N \left(\frac{h}{2\pi} \right)^2 \Delta^2 + V(r, R)$$

and energy E is obtained for internuclear separation R , which becomes varied to obtain $E(R)$ that acts as a potential energy for the motion of the atomic nuclei.

The matrix named *Occupation* defines the state of initial occupation of the molecular orbitals. In the section of parameters and auxiliary functions are defined

- the squared distance between two functions
- the product of two gaussians

– auxiliary function AuxInt, which not zero corresponds to $\frac{\sqrt{\frac{\pi}{x}} \operatorname{erf}(x)}{2}$

- a procedure to find the eigenvalues and eigenvectors of a diagonalizable matrix, and to order the eigenvalues from small to large

In the section on integral functions are defined the forms to evaluate the integrals in question.

$$\langle \phi | \phi \rangle = \text{overlap integral} = Q aux^{\frac{3}{4}}$$

in which

$$Q = e^{-\frac{\alpha_1 \alpha_2}{\alpha_1 + \alpha_2} \sqrt{(\text{func1} - \text{func2})^2}} \quad \text{and} \quad aux = \frac{4 \alpha_1 \alpha_2}{(\alpha_1 + \alpha_2)^2}$$

The integral for kinetic energy, $\langle \phi | T | \phi \rangle$, is

$$\langle \phi | \phi \rangle E \left(3 - 2 E \sqrt{(\text{func1} - \text{func2})^2} \right) \quad \text{and} \quad E = \frac{\alpha_1 \alpha_2}{\alpha_1 + \alpha_2}$$

The integral for potential energy $\langle \phi | V | \phi \rangle$ is

$$- \langle \phi | \phi \rangle \frac{2 \sqrt{c} V}{\sqrt{\pi}}$$

in which V is calculated in a recursive form beginning with $V=0$, according to

$$V = V + \text{AtomicNumber}_{nuc} * \text{AuxInt}(arg)$$

with

$$arg = c \left((Rx - \text{NuclearCoordinate}_{nuc, X})^2 + (Ry - \text{NuclearCoordinate}_{nuc, Y})^2 + (Rz - \text{NuclearCoordinate}_{nuc, Z})^2 \right)$$

On invoking procedure ProductGaussian, the four-centre integrals become those of two centres.

$$\langle \phi \phi | 1/r | \phi \phi \rangle = \frac{2 \text{OverlapIntegral}_{i,j} \text{OverlapIntegral}_{k,l} \sqrt{\frac{c_1 c_2}{c_1 + c_2}} \text{AuxInt}(arg)}{\sqrt{\pi}}$$

in which

$$arg =$$

$$\frac{c_1 c_2 ((Rx1 - Rx2)^2 + (Ry1 - Ry2)^2 + (Rz1 - Rz2)^2)}{c_1 + c_2}$$

MOperator takes into account two-electron integrals of a filled shell, working in a recursive manner according to

$$sumd := sumd + DensityMatrix[k,l]*(TwoElectronIntegral(i,j,k,l)),$$

this being the Coulomb integral, not considering interchange.

ρ , which is involved in the calculation of $V_{XC}[\rho] = \delta E_{XC}[\rho]/\delta \rho$, is evaluated in a recursive manner

according to $\rho := \left(\frac{2}{\pi}\right)^{\frac{3}{2}} sumd$, in which

$$sumd = sumd + 2 \left(\alpha_{row} \alpha_{column}\right)^{\frac{3}{4}} DensityMatrix_{row, column} e^{-(\alpha_{row} + \alpha_{column})r^2}$$

beginning with $sumd = 0$. Value of X pertains to the exchange functional that is used for the integration of trapezoids, in procedure *trapzd*, and is defined as

$$x^2 e^{-\frac{exponent_i + exponent_j x^2 + \ln(\rho(x))}{3}}$$

Then $V_{XC}(\rho) = \frac{\delta E_{XC}(\rho)}{\delta \rho}$ with $E_{XC} = \int \rho(r) \in dr$, which is realized by means of

$$evalf\left(-\left(\frac{2}{\pi}\right)^{\frac{3}{2}} \cdot 4\pi \left(\frac{3}{\pi}\right)^{\frac{1}{3}} \sqrt{\sqrt{(Exponent_{ro} Exponent_{colm})^3} DensityIntegral(ro, colm)}\right)$$

The construction of the matrices is realized on applying the integral functions defined above for each matrix element.

The solution of the problem of diagonalization has two forms, with or without considering electronic correlation. Without such consideration, the steps are

- i) diagonalization of Smatrix = overlap matrix: $\lambda(B^T, S, B) = (D = SSpur)$,
- ii) finding the square roots of the elements of D (RootSMatrix),
- iii) construction the transformation matrix $TransMatrix = B \cdot RootSMatrix^{-1}$,
- iv) forming the Fock operator, $F = Hcore = TMatrix + VMatrix = TVMatrix$,
- v) constructing the Fock operator transformed to an orthonormal basis set: $FockOp = TransMatrix^T \cdot TVMatrix \cdot TransMatrix$
- vi) diagonalizing FockOp by means of the procedure to find eigenvalues and eigenvectors, $\lambda(c^{-1}, FockOp, c) = E$, in which c is the matrix of orthonormal eigenvectors and E has on its diagonal the eigenvalues as energies,
- vii) to obtain the coefficients of the original basis set, the obtained eigenvectors are retransformed; the new vectors represent the coefficients for those that multiply the gaussian functions.

Additional steps of density-functional theory follow.

- i) Matrix MMatrix is constructed of which elements $M_{\mu, \nu} = \text{element } \mu \nu \text{ of Matrix}(2J) =$

$$\sum_{\lambda} \sum_{\rho} \frac{P_{\lambda \rho}(\mu \nu)}{\lambda \rho} \text{ in which } P_{\lambda \rho} \text{ is the density matrix.}$$

- ii) Matrix VxMatrix is formed using the exchange operator.

- iii) A new Fock operator is formed as $F = H_{core} + MMatrix$, which is transformed to an orthonormal

basis set.

iv) The energies and the coefficients to calculate the self-consistent field are found, for which purpose one clicks on the link "NEXT DFT STEP" until convergence.

input (nuclear data, basis-set data, occupation matrix)

> restart:

> Digits := 12:

> NumberofNuclei := 1: Example: helium atom i.e. 1 nucleus at the origin

> Dim := 3: 3 basis functions on the nucleus

definition of nuclear data -- atomic number, $X Y Z$

> AtomicNumber := <2>;

$$AtomicNumber := \begin{bmatrix} 2 \end{bmatrix} \quad (18.3.1)$$

> NuclearCoordinate := <<0|0|0>>;

$$NuclearCoordinate := \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \quad (18.3.2)$$

definition of basis-set data -- exponents, $X Y Z$

> Exponent := <3.6,1.2,0.4>;

$$Exponent := \begin{bmatrix} 3.6 \\ 1.2 \\ 0.4 \end{bmatrix} \quad (18.3.3)$$

> BasisCoordinate := <<0|0|0>,<0|0|0>,<0|0|0>>;

$$BasisCoordinate := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (18.3.4)$$

occupation matrix in the molecular-orbital representation (MO-basis)

> Occupation := <<2|0|0>,<0|0|0>,<0|0|0>>; lowest MO occupied with 2 electrons

$$Occupation := \begin{bmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (18.3.5)$$

parameters and auxiliary functions

procedure to calculate the square of the distance between two basis functions

> X := 1:

Y := 2:

Z := 3:

These values serve as indices to make the program more readable, because x , y and z -coordinates are stored in arrays.

> sqrDistance := proc (func1, func2)

(BasisCoordinate[func1,X] - BasisCoordinate[func2,X])^2 +

```

    (BasisCoordinate[func1,Y] - BasisCoordinate[func2,Y])^2 +
    (BasisCoordinate[func1,Z] - BasisCoordinate[func2,Z])^2;
end proc:

```

calculation of the product of two gaussian functions

The product of two gaussian functions is also a gaussian function with a new exponent *cnew*, which is the sum of the original exponents, and new coordinates *Rxnew*, *Rynew*, *Rznew*, which are found from the old ones by weighting with the exponents.

```

> ProductGaussian := proc(func1,func2,cnew::evaln,Rxnew::evaln,
                          Rynew::evaln,Rznew::evaln)

    local a,b;
    cnew := Exponent[func1] + Exponent[func2];
    a := Exponent[func1] / eval(cnew);
    b := Exponent[func2] / eval(cnew);
    Rxnew := a*BasisCoordinate[func1,X] + b*BasisCoordinate[func2,X]
;
    Rynew := a*BasisCoordinate[func1,Y] + b*BasisCoordinate[func2,Y]
;
    Rznew := a*BasisCoordinate[func1,Z] + b*BasisCoordinate[func2,Z]
;
end proc:

```

auxiliary procedure for calculation of the potential and two-electron-integrals

```

> rootPi := sqrt(Pi):
AuxInt := proc(X)
    local rootX;
    if X=0
        then 1
        else
            rootX := sqrt(X);
            evalf(1/2*rootPi/rootX*erf(rootX));
        end if;
    end proc:

```

procedure to form and to sort eigenvalues and eigenvectors of a real matrix made symmetric

A matrix to be diagonalized is input, and eigenvalues and eigenvectors of the symmetrized matrix are returned sorted in ascending order.

```

> Seig := proc(A::Matrix)
    local A1,evalu,evect,evalul,ss,ev2,ev3,ev4,
          evecl,evecl1,evecl2,evecM,NrA;
    A1 := 1/2*(A + LinearAlgebra:-Transpose(A));
    NrA := LinearAlgebra:-RowDimension(A1);
    evalu, evect := evalf(LinearAlgebra:-Eigenvectors(A1,
        output=['values', 'vectors']));
    evalul := [seq([Re(evalu[i]), i], i=1..NrA)];

```

```

ss := proc(a,b) evalb(a[1] < b[1]); end proc:
ev2 := sort(evalul, ss);
ev3 := map2(op, 1, ev2);
ev3 := [seq(op(1, ev2[j]), j=1..NrA)];
ev4 := map2(op, 2, ev2);
ev4 := [seq(op(2, ev2[j]), j=1..NrA)];
evecl1 := convert(
    LinearAlgebra:-Transpose(map(Re, vect)), listlist);
evecl := [seq(evecl1[ev4[i]], i=1..NrA)];
evecl2 := map(proc(a) map(Re, a) end proc, evecl);
evecM := LinearAlgebra:-Transpose(Matrix(evecl2));
convert(ev3, Vector), evecM;
end proc:

```

Warning, (in Seig) `i` is implicitly declared local
Warning, (in Seig) `j` is implicitly declared local

integral functions -- S, T, V, 2e-integral, M and Vx Operators

scf-like functions

procedure to calculate the overlap integrals $\langle \phi | \phi \rangle$

```

> Overlapintegral := proc (func1, func2)
    local alpha, beta, cinv, Q, aux;
    alpha := Exponent[func1];
    beta := Exponent[func2];
    cinv := 1/(alpha + beta);
    Q := exp(-alpha * beta * cinv * sqrDistance(func1,func2));
    aux := (4*alpha*beta*cinv^2);
    Q*sqr(sqrt(aux^3));
end proc:

```

procedure to calculate the kinetic-energy integral $\langle \phi | T | \phi \rangle$

```

> KineticIntegral := proc (func1, func2)
    local alpha, beta, E;
    alpha := Exponent[func1];
    beta := Exponent[func2];
    E := alpha * beta / (alpha+beta);
    Overlapintegral(func1,func2)*E*(3-2*E*sqrDistance(func1,func2));
end proc:

```

procedure to calculate the potential-energy integral $\langle \phi | V | \phi \rangle$

```

> PotentialIntegral := proc(func1, func2)
    local c,Rx,Ry,Rz,V,argument,nuc;
    global NumberofNuclei;
    ProductGaussian(func1,func2,c,Rx,Ry,Rz);
    V := 0;

```

```

for nuc to NumberofNuclei do
    argument := c*((Rx - NuclearCoordinate[nuc,X])^2 +
        (Ry - NuclearCoordinate[nuc,Y])^2 +
        (Rz - NuclearCoordinate[nuc,Z])^2);
    V := V + AtomicNumber[nuc] * AuxInt(argument);
end do;
evalf(-Overlapintegral(func1,func2) * 2/sqrt(Pi) * sqrt(c) * V);
end proc:

```

two-electron integral $\langle \phi\phi | 1/r | \phi\phi \rangle$

```

> TwoElectronIntegral := proc(i,j,k,l)
    local Argumen, cnew, c1, c2, Rx1, Rx2, Ry1, Ry2, Rz1, Rz2;
    ProductGaussian(i,j,c1,Rx1,Ry1,Rz1);
    ProductGaussian(k,l,c2,Rx2,Ry2,Rz2);
    cnew := c1 * c2 / (c1 + c2);
    Argumen := cnew*((Rx1-Rx2)^2 + (Ry1-Ry2)^2 + (Rz1-Rz2)^2);
    evalf(2/sqrt(Pi)) * Overlapintegral(i, j) * Overlapintegral(k,
1)
        * sqrt(cnew) * AuxInt(Argumen);
end proc:

```

two-electron operator for closed shells

This operator works as for a self-consistent field, but without an exchange integral.

```

> MOperator := proc(i,j)
    local sumd, k, l;
    global Dim;
    sumd := 0;
    for k to Dim do
        for l to Dim do
            sumd := sumd + DensityMatrix[k,l]*TwoElectronIntegral(i,j,k,
1);
        end do;
    end do;
    sumd;
end proc:

```

exchange functional

rho - calculates the density, called by ValueOfVx

```

> rho := proc(r)
    local ro, colm, sumd, mu, nu, fac, pref;
    global Dim;
    pref := evalf((2/Pi)^(3/2), 18);
    sumd := 0;
    for ro to Dim do

```

```

        for colm to ro do
            mu := Exponent[ro];
            nu := Exponent[colm];
            fac := 2.0*(mu * nu)^(3/4);
            if ro = colm then
                fac := fac/2;
            end if;
            sumd := sumd + fac*DensityMatrix[ro,colm]*exp(-(mu+nu)*r^2)
        ;
    end do;
end do;
pref*sumd;
end proc:

```

ValueOfVx - calculates the value of the exchange functional

This procedure uses rho and is called by the numerical integration *trapzd*, which in turn is called by *DensityIntegral*.

```

> ValueOfVx := proc(i,j,x)
    local rhotmp,result;
    rhotmp := rho(x);
    if rhotmp = 0 then
        result := 0;
    else
        result := x^2*exp(-(Exponent[i]+Exponent[j])*x^2+ln(rhotmp)/3)
    ;
    end if;
    result;
end proc:

```

numerical integration over exchange density, for only a spherical density

procedure for trapezoidal integration used in *DensityIntegral*

```

> trapzd := proc(i,j)
    local k,n,os,s,w,x;
    n := 2^7;
    w := 10.0/n;
    s := 0;
    for k from 1 to n-1 do
        os := s;
        s := s + ValueOfVx(i,j,k*w);
        if s = os then break end if;
    end do;
    # Assume end points to be negligible.
    # Assume a unimodal function ValueOfVx.
    s*w;

```



```
end proc:
```

DensityIntegral - calculates the integral used in VxOperator

```
> DensityIntegral := proc(i,j)
```

```
    trapzd(i,j);
```

```
end proc:
```

exchange operator

```
> VxOperator := proc(ro,colm)
```

```
    local prefac;
```

```
    prefac := evalf(-(2/Pi)^(3/2)*4*Pi*(3/Pi)^(1/3), 18);
```

```
    evalf(prefac * sqrt(sqrt(((Exponent[ro] * Exponent[colm])^3)))
```

```
        *DensityIntegral(ro,colm), 18);
```

```
end proc:
```

construction of integral matrices

same as in SCF program

overlap matrix S

```
> SMatrix := Matrix(1..Dim,1..Dim):
```

```
for ro to Dim do
```

```
    for colm to ro do
```

```
        SMatrix[ro,colm] := Overlapintegral(ro,colm);
```

```
        SMatrix[colm,ro] := SMatrix[ro,colm];
```

```
    end do:
```

```
end do:
```

```
> print (SMatrix):
```

$$\begin{bmatrix} 1.000000000000 & 0.805927448864 & 0.464758001545 \\ 0.805927448864 & 1.000000000000 & 0.805927448867 \\ 0.464758001545 & 0.805927448867 & 1.000000000000 \end{bmatrix}$$

(18.6.1.1)

kinetic-energy matrix T

```
> TMatrix := Matrix(1..Dim,1..Dim):
```

```
for ro to Dim do
```

```
    for colm to ro do
```

```
        TMatrix[ro,colm] := KineticIntegral(ro,colm);
```

```
        TMatrix[colm,ro] := TMatrix[ro,colm];
```

```
    end do:
```

```
end do:
```

```
> print(TMatrix);
```

$$\begin{bmatrix} 5.400000000000 & 2.17600411193 & 0.501938641668 \\ 2.17600411193 & 1.800000000000 & 0.725334703980 \\ 0.501938641668 & 0.725334703980 & 0.600000000000 \end{bmatrix}$$

(18.6.2.1)

potential-energy matrix V

```
> VMatrix := Matrix(1..Dim,1..Dim):
  for ro to Dim do
    for colm to ro do
      VMatrix[ro,colm] := PotentialIntegral(ro,colm);
      VMatrix[colm,ro] := VMatrix[ro,colm];
    end do:
  end do:
> print(VMatrix);
```

$$\begin{bmatrix} -6.05551805284 & -3.98475497212 & -2.09769298672 \\ -3.98475497212 & -3.49615497788 & -2.30059935582 \\ -2.09769298672 & -2.30059935582 & -2.01850601762 \end{bmatrix} \quad (18.6.3.1)$$

DFT step -- solution of the general eigenvalue problem

same as in SCF program

orthogonalisation of the basis set

The general eigenvalue problem can be simplified to the special eigenvalue problem through a transformation to an orthonormal basis set; this result is obtained on diagonalisation. Othonormalization signifies that the overlap matrix must be transformed to the unit matrix, i.e. a transformation matrix **A** has to be found such that **A^T S A = I**. For this purpose we first diagonalize **B^T S B = D**, in which **B** = "EigenVectors" is the transformation matrix, and **D** = "SSpur" is diagonal.

```
> SSpur := simplify(LinearAlgebra:-DiagonalMatrix(
      LinearAlgebra:-Eigenvalues(SMatrix))):
> EValues,EVectors := evalf(Seig(SMatrix));
```

$$EValues, EVectors := \begin{bmatrix} 0.0691773391206041 \\ 0.535241998455000 \\ 2.39558066242440 \end{bmatrix}, \quad (18.7.1.1)$$

$$\begin{bmatrix} 0.447276387071647 & -0.707106781188519 & -0.547671282399712 \\ -0.774524155292659 & 2.27414675020518 \times 10^{-12} & -0.632544332729487 \\ 0.447276387076633 & 0.707106781184576 & -0.547671282400732 \end{bmatrix}$$

We form the square-root of the diagonal matrix = "RootSMatrix".

```
> RootSMatrix := SSpur:
  for i to Dim do
    RootSMatrix[i,i] := sqrt(EValues[i]);
  end do:
  print(RootSMatrix);
```

$$\begin{bmatrix} 0.263015853363641 & 0. & 0. \\ 0. & 0.731602349951803 & 0. \\ 0. & 0. & 1.54776634619842 \end{bmatrix} \quad (18.7.1.2)$$

Multiplying inverse RootSMatrix **W** with the eigenvectors **B** from the left yields the desired transformation matrix:

$$\mathbf{A} = \mathbf{B} \mathbf{W}, \text{ as } (\mathbf{B} \mathbf{W})^T \mathbf{S} \mathbf{B} \mathbf{W} = \mathbf{W} (\mathbf{B}^T \mathbf{S} \mathbf{B}) \mathbf{W} = \mathbf{W} \mathbf{D} \mathbf{W} = \mathbf{I}$$

```
> TransMatrix := EVectors.LinearAlgebra:-MatrixInverse(RootSMatrix)
;
```

TransMatrix := (18.7.1.3)

$$\begin{bmatrix} 1.70056816481420 & -0.966517919516118 & -0.353846227335856 \\ -2.94478125705152 & 3.10844648100843 \times 10^{-12} & -0.408682056101766 \\ 1.70056816483316 & 0.966517919510728 & -0.353846227336515 \end{bmatrix}$$

Fock operator, transformation to the orthogonal basis set and diagonalization

In the first SCF-step, the density matrix is set to zero, i.e. the problem is treated as one-electron problem. The Fock operator in a one-electron problem is $F = H_{\text{core}} = T + V = \text{TVMatrix}$. Transformation to the orthonormal basis set yields $F' = \text{ATFA} = \text{AT TVMatrix A}$.

```
> TVMatrix := simplify(TMatrix + VMatrix); Hcore = T + V
```

$$\text{TVMatrix} := \begin{bmatrix} -0.655518052840 & -1.80875086019 & -1.59575434505 \\ -1.80875086019 & -1.69615497788 & -1.57526465184 \\ -1.59575434505 & -1.57526465184 & -1.41850601762 \end{bmatrix} \quad (18.7.2.1)$$

```
> FockOp := simplify(LinearAlgebra:-Transpose(TransMatrix).TVMatrix.
TransMatrix);
```

$$\text{FockOp} := \begin{bmatrix} 3.95678186277 & -1.91861428324 & -0.0470882072422 \\ -1.91861428324 & 1.04390595550 & 0.168714197710 \\ -0.0470882072422 & 0.168714197710 & -1.92130495587 \end{bmatrix} \quad (18.7.2.2)$$

The Fock operator F' is diagonalised and yields as eigenvalues the orbital energies and as eigenvectors the coefficients c' representing the molecular orbitals in the orthonormal basis set.

```
> EValues,OBEigenVec := Seig(FockOp);
```

$$\text{EValues, OBEigenVec} := \begin{bmatrix} -1.93168926881681 \\ 0.0999223394691459 \\ 4.91114979174766 \end{bmatrix}, \quad (18.7.2.3)$$

$$\begin{bmatrix} -0.0132346904050450 & -0.445099484938745 & 0.895383320973284 \\ -0.0651075406864903 & -0.893177871602819 & -0.444965501836733 \\ 0.997790484578621 & -0.0641851866580076 & -0.0171584002178849 \end{bmatrix}$$

back transformation of the coefficients and formation of the density matrix

back transformation of the coefficients to the original AO Basis: $c = \text{Ac}'$

```
> MOEigenVectors := TransMatrix.OBEigenVec;
```

MOEigenVectors := (18.7.3.1)

$$\begin{bmatrix} -0.312643287044635 & 0.129062090107147 & 1.95879893722868 \\ -0.368805798548915 & 1.33695195482279 & -2.62969569119959 \\ -0.438498496583427 & -1.59748274633522 & 1.09866467506470 \end{bmatrix}$$

transformation of the occupation matrix into the MO basis (input) into the natural AO basis --> density matrix

```
> DensityMatrix := MOEigenVectors.Occupation.LinearAlgebra:-
  Transpose(MOEigenVectors);
```

$$DensityMatrix := \begin{bmatrix} 0.195491649868147 & 0.230609314278908 & 0.274187222671946 \\ 0.230609314278908 & 0.272035434086606 & 0.323441576389899 \\ 0.274187222671946 & 0.323441576389899 & 0.384561863011852 \end{bmatrix} \quad (18.7.3.2)$$

further DFT steps

two-electron coulombic-operator matrix At this point, the calculation diverges from the posted version.

$M_{\mu\nu}$ = Element $\mu\nu$ of matrix **M** = Element $\mu\nu$ of the 2J-Matrix = $\sum_{\lambda} \sum_{\rho} P_{\lambda\rho} (\mu\nu/\lambda\rho)$

```
> MMatrix := Matrix(1..Dim,1..Dim):
```

```
  for ro to Dim do
```

```
    for colm to ro do
```

```
      MMatrix[ro,colm] := MOperator(ro,colm);
```

```
      MMatrix[colm,ro] := MMatrix[ro,colm];
```

```
    end do:
```

```
  end do:
```

```
> print(MMatrix);
```

$$\begin{bmatrix} 3.04459007006438 & 2.28958718277532 & 1.27302014055833 \\ 2.28958718277532 & 2.42746650891774 & 1.74330630160891 \\ 1.27302014055833 & 1.74330630160891 & 1.71112753990624 \end{bmatrix} \quad (18.8.1.1)$$

two-electron exchange-operator matrix

```
> VxMatrix := Matrix(1..Dim,1..Dim):
```

```
  for ro to Dim do
```

```
    for colm to ro do
```

```
      VxMatrix[ro,colm] := VxOperator(ro,colm);
```

```
      VxMatrix[colm,ro] := VxMatrix[ro,colm];
```

```
    end do:
```

```
  end do:
```

```
> print(VxMatrix);
```

$$\begin{bmatrix} -0.970374846415247483 & -0.705609727195862267 & -0.385226219292929860 \\ -0.705609727195862267 & -0.690243601982749891 & -0.465848273893789350 \\ -0.385226219292929860 & -0.465848273893789350 & -0.397671031196354725 \end{bmatrix} \quad (18.8.2.1)$$

Fock operator

$$F = H_{core} + M + Vx$$

```
> FMatrix := TVMatrix + MMatrix + VxMatrix;
```

$$FMatrix := \begin{bmatrix} 1.41869717080914 & -0.224773404610542 & -0.707960423784596 \\ -0.224773404610542 & 0.0410679290549917 & -0.297806624124877 \\ -0.707960423784596 & -0.297806624124877 & -0.105049508910112 \end{bmatrix} \quad (18.8.3.1)$$

$F' = A^T F A$, transformation of the Fock operator to the orthonormal basis set

```
> FockOp := LinearAlgebra:-Transpose(TransMatrix).FMatrix.
TransMatrix;
```

$$FockOp := \begin{bmatrix} 5.29431700939630 & -2.29660918257844 & -0.0703732439253514 \\ -2.29660918257844 & 2.54984522641653 & 0.549967427657240 \\ -0.0703732439253515 & 0.549967427657240 & -0.157087353750073 \end{bmatrix} \quad (18.8.3.2)$$

```
> EValues, OBEigenVec := Seig(FockOp);
```

$$EValues, OBEigenVec := \begin{bmatrix} -0.300324591384140 \\ 1.37354990495173 \\ 6.61384956849516 \end{bmatrix}, \quad (18.8.3.3)$$

$$\begin{bmatrix} -0.0956055422707988 & -0.489637321489747 & 0.866668837383376 \\ -0.262323000417873 & -0.827487185418784 & -0.496438920129620 \\ 0.960232380072069 & -0.274809481954399 & -0.0493307702056612 \end{bmatrix}$$

The Fock operator F' is diagonalised and yields as eigenvalues the orbital energies and as eigenvectors the coefficients c' representing the molecular orbitals in the orthonormal basis set.

calculation of density matrix

back transformation of the coefficients to the original AO basis: $c = A c'$

```
> MOEigenVectors := TransMatrix.OBEigenVec;
```

$$MOEigenVectors := \begin{bmatrix} -0.248818466014642 & 0.0643598500725369 & 1.97110205347006 \\ -0.110892334474867 & 1.55418451119473 & -2.53198954780201 \\ -0.755898227225890 & -1.53520253568653 & 1.01146782898818 \end{bmatrix} \quad (18.8.4.1)$$

transformation of the occupation matrix from the MO basis (see input) to the original AO basis -> density matrix P

```
> DensityMatrix := MOEigenVectors.Occupation.LinearAlgebra:-
Transpose(MOEigenVectors);
```

$$DensityMatrix := \begin{bmatrix} 0.123821258059759 & 0.0551841211136378 & 0.376162874723066 \\ 0.0551841211136378 & 0.0245942196905714 & 0.167646638084984 \\ 0.376162874723066 & 0.167646638084984 & 1.14276425984649 \end{bmatrix} \quad (18.8.4.2)$$

convergence

energy = trace of "EMatrix" = trace $[P(\mathbf{H} + 1/2 \mathbf{M})]$

```
> EMatrix := 1/2*DensityMatrix.(TVMatrix + 1/2*VxMatrix + FMatrix);  
EMatrix :=  $\begin{bmatrix} -0.518146032641889 & -0.599027665459134 & -0.536607765089149 \\ -0.230925075935354 & -0.266972051157504 & -0.239153020759219 \\ -1.57410209053810 & -1.81981650169476 & -1.63018792312089 \end{bmatrix}$  (18.8.5.1)
```

The energy is saved in Etot_old, for a comparison in the next step.

```
> Etot_old := Etot;  
Form a trace.  
> Etot := LinearAlgebra:-Trace(EMatrix);  
Etot := -2.41530600692028 (18.8.5.2)
```

Calculate the energy decrease in the last SCF step toward convergence.

```
> Dif := Etot - Etot_old;  
Dif := 0. (18.8.5.3)
```

To undertake the next iteration, click on **NEXT DFT STEP** below and execute the statements beginning with that indicated by the cursor; repeat this iterative process until a satisfactory convergence is achieved such that **Dif** is acceptably small.

NEXT DFT STEP

e3.81 exercise

Develop two or three exercises for each of these three sets of procedures. If possible, please convert all linalg calls to LinearAlgebra calls in "one-electron program for small molecules". You can verify the correctness of the calculations at www.chemie.unibas.ch/~huber/DFT/DFT.html or others at www.chemie.unibas.ch/~huber.

3.9 quaternions

3.91 introduction to quaternions

```
> restart;  
with(LinearAlgebra):
```

A complex number z has a form $z = a + b i$ in which a, b are real numbers and $i = \sqrt{-1}$. A

hypercomplex quantity has an extended form $q = a + \sum_{j=1}^{j_{\max}} b_j e_j$ in which a and b_j are real numbers and e_j

are generalizations of i . If $j_{\max} = 3$, the hypercomplex quantity is a quaternion, whereas $j_{\max} = 7$ for an octonion or Cayley number, et cetera; octonions fail generally not only to commute in multiplication but also to be associative, $p(q*r) \neq (p*q)r$. A theorem states that the only normed division algebras, which are number systems in which one can add, subtract, multiply and divide, and which have a norm satisfying $|z w| = |z| |w|$, have dimension 1, 2, 4 or 8: dimension 1 implies the domain of real number; dimension 2 implies the domain of complex number; dimension 4 implies the domain of quaternion; dimension 8 implies the domain of octonion, which might be regarded as octets of real numbers and which are neither commutative nor associative. Both quaternions and octonions have served as a basis

of quantum mechanics; Wigner and von Neumann tried, but failed, to make octonions the basis of quantum mechanics, but quaternions have served this purpose. Both octonions over real numbers and quaternions over complex numbers have eight components, but these quantities are distinct. Objects related to quaternions arise from the solution of the Dirac equation for the electron; their non-commutativity is essential in that application.

Described by A. R. Hamilton in 1843, a *quaternion* originated before a [vector](#), was a forerunner of the study of [matrices](#) and serves as a way to generalise a complex number defined or depicted in a [plane](#), such as in an [Argand diagram](#), to three [dimensions](#); it comprises four components such that, for quaternion q ,

$$q = q_0 \mathbf{1} + q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k}$$

in which four coefficients x_r , $0 \leq r \leq 3$, are real numbers, $\mathbf{1}$ is just unity, and $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are hypercomplex numbers and elements of a basis; the laws of multiplication for the latter quantities are

$$\begin{aligned} \mathbf{1} * \mathbf{1} &= \mathbf{1}, & \mathbf{1} * \mathbf{i} &= \mathbf{i}, & \mathbf{1} * \mathbf{j} &= \mathbf{j}, & \mathbf{1} * \mathbf{k} &= \mathbf{k}, \\ \mathbf{i} * \mathbf{1} &= \mathbf{i}, & \mathbf{i} * \mathbf{i} &= -\mathbf{1}, & \mathbf{i} * \mathbf{j} &= \mathbf{k}, & \mathbf{i} * \mathbf{k} &= -\mathbf{j}, \\ \mathbf{j} * \mathbf{1} &= \mathbf{j}, & \mathbf{j} * \mathbf{i} &= -\mathbf{k}, & \mathbf{j} * \mathbf{j} &= -\mathbf{1}, & \mathbf{j} * \mathbf{k} &= \mathbf{i}, \\ \mathbf{k} * \mathbf{1} &= \mathbf{k}, & \mathbf{k} * \mathbf{i} &= \mathbf{j}, & \mathbf{k} * \mathbf{j} &= -\mathbf{i}, & \mathbf{k} * \mathbf{k} &= -\mathbf{1}, \end{aligned}$$

so $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{i} \mathbf{j} \mathbf{k} = -1$, as Hamilton inscribed in the stonework of a bridge in Dublin in 1843. The product of two quaternions resembles the [cross](#) or vectorial product of vectors. The set of all quaternions is regarded as a vector space of dimension four over the field of real numbers in which $\mathbf{1} = (1,0,0,0)$, $\mathbf{i} = (0,1,0,0)$, $\mathbf{j} = (0,0,1,0)$, $\mathbf{k} = (0,0,0,1)$ are the symbols used to denote the standard basis vectors of real vector space \mathbf{R}^4 . The unit quaternion is $(1,0,0,0)$; the zero quaternion is $(0,0,0,0)$. The conjugate of quaternion $q = q_0 \mathbf{1} + q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k}$ is $q^* = q_0 \mathbf{1} - q_1 \mathbf{i} - q_2 \mathbf{j} - q_3 \mathbf{k}$, so that $|q|^2 = q q^* = q^* q = q_0^2 + q_1^2 + q_2^2 + q_3^2$. If a quaternion be complex, i.e. have complex numbers as coefficients of the elements of the basis, its conjugate is $q^* = q_0^* \mathbf{1} + q_1^* \mathbf{i} + q_2^* \mathbf{j} + q_3^* \mathbf{k}$. The conjugate of a product $q r$ of two quaternions q and r is $(q r)^* = q^* r^*$.

A quaternion is a vector in a real vector space of four dimensions, but can also be defined as a union of a real scalar and a real three-dimensional vector, in which $\mathbf{1}$ = real number unity is the scalar part, and $\mathbf{i} = (1,0,0)$, $\mathbf{j} = (0,1,0)$, $\mathbf{k} = (0,0,1)$ form the vector part. Quaternions form a vector space under operations addition of vectors and scalar multiplication of a real scalar with a vector. When we include a third operation quaternion multiplication that is defined by [bilinearity](#) and the multiplication table above, they form a real, associative, non-commutative [division algebra](#); Frobenius proved an important theorem that quaternions form the only non-[commutative associative](#) division algebra over [real numbers](#). Quaternions thus combine according to conventional algebraic laws except that multiplication is non-commutative but conforms to the above laws; equality, sum and scalar multiple are thus defined component by component. If $q_1 = x$, $q_2 = y$, $q_3 = z$, $q_0 = i c t$, then $q q^* = x^2 + y^2 + z^2 - c^2 t^2 = s^2$, which is a relativistic invariant: a quaternion can represent a relativistic *four-vector*, which is a vector with four components.

A quaternion resembles a vector in four dimensions endowed with a rule for multiplication that is associative but not commutative, distributes through addition, contains an identity, and, most crucially, for which each nonzero vector in four dimensions has a unique multiplicative inverse. The quaternions hence form a division algebra. Ever since the discovery of quaternion multiplication, quaternions have been applied to rotate vectors in three dimensions on sandwiching a vector between a unit quaternion and its conjugate. A quaternion is represented with just four scalars, in contrast to a rotation matrix of third order that has nine scalar entries.

For a spatial vector represented with $x \mathbf{i} + y \mathbf{j} + z \mathbf{k}$, the general quaternion $q = a + b \mathbf{i} + c \mathbf{j} + d \mathbf{k}$ represents vector f in four dimensions; its conjugate is $q^* = a - b \mathbf{i} - c \mathbf{j} - d \mathbf{k}$ and the square of the

magnitude of f is $|q|^2 = q^* q = a^2 + b^2 + c^2 + d^2$, as above, as an extension of the rule for complex numbers. A multiplication of quaternions, $p q$, becomes an operation equivalent to conversion of four-vector q into another four-vector represented by product $p q$; for

$$q = a + b \mathbf{i} + c \mathbf{j} + d \mathbf{k} \quad \text{and} \quad p = \alpha + \beta \mathbf{i} + \gamma \mathbf{j} + \delta \mathbf{k},$$

the product,

$$p q = (\alpha a - \beta b - \gamma c - \delta d) + (\alpha b + \beta a + \gamma d - \delta c) \mathbf{i} + (\alpha c - \beta d + \gamma a + \delta d) \mathbf{j} + (\alpha d + \beta c - \gamma b + \delta a) \mathbf{k}$$

is another quaternion that can represent a four-vector. According to the rules of multiplication, not commutative, and expansion of an exponential function with $\alpha^2 + \beta^2 + \gamma^2 = 1$,

$$e^{\theta(\alpha \mathbf{i} + \beta \mathbf{j} + \gamma \mathbf{k})} = \cos(\theta) + \sin(\theta) (\alpha \mathbf{i} + \beta \mathbf{j} + \gamma \mathbf{k})$$

which is the analogue of [Euler's formula](#) for the imaginary exponential function. The exponent represents a vector in space of length θ and of direction given by the [direction cosines](#) α, β, γ whereas the exponential formula itself has unit length. Any quaternion can be represented in form

$Q e^{\theta(\alpha \mathbf{i} + \beta \mathbf{j} + \gamma \mathbf{k})}$, in which scalar multiplicand Q is the length of the four-vector and θ and the direction cosines fix the direction in four-space. With $Q = 1$, this exponential operator is related to a rotational operator. The most general rotation of a four-vector represented with $q = w + x \mathbf{i} + y \mathbf{j} + z \mathbf{k}$ is

$$q' = e^{\theta(\alpha \mathbf{i} + \beta \mathbf{j} + \gamma \mathbf{k})} q e^{-\phi(\lambda \mathbf{i} + \mu \mathbf{j} + \nu \mathbf{k})}$$

in which $\alpha^2 + \beta^2 + \gamma^2 = \lambda^2 + \mu^2 + \nu^2 = 1$. When both θ and ϕ are imaginary angles, the transformation corresponds to a [Lorentz transformation](#).

The applications of quaternions are concerned primarily with rotations, such as for rotation about an arbitrary axis, but they serve also as alternative to Cayley-Klein parameters in a treatment of angular momentum [cf. Whittaker and Watson, *Classical Mechanics*], including particle spin [cf. S. L. Altmann, *Rotations, Quaternions and Double Groups*, Oxford, 1986]. Quaternions are well adapted to treat four-dimensional euclidean space, as introduced above, and are considered to be a particular example of a Clifford algebra used in theoretical physics in connexion with supersymmetries, implying a space of higher dimension [cf. J. S. R. Chisholm and A. K. Common, *Clifford Algebras and their Applications in Mathematical Physics*, Reidel, 1985]. The real algebra of quaternions is isomorphic to the real linear span of the Pauli spin matrices; notice the reversed signs with multiplicand $I = \sqrt{-1}$ or the reverse order of these spin matrices with the representation of a quaternion just below.

$$1 \rightarrow \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \mathbf{i} \rightarrow \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \mathbf{j} \rightarrow \begin{bmatrix} 0 & -I \\ I & 0 \end{bmatrix}, \mathbf{k} \rightarrow \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Other applications relate to lattice dynamics of rigid molecules including anharmonicity, in which displacement coordinates based on quaternions allow proper conjugate momenta to be defined [cf. S. H. Walmsley, *J. Chim. Phys.* **82** (1985) 117, *J. Mol. Structure* **189** (1988) 129, *Int Rev Phys Chem.* **5** (1985) 185]. Quaternions in quantum mechanics are discussed by S. L. Adler [*Quaternionic Quantum Mechanics and Quantum Fields*, Oxford, 1995].

A quaternion can be represented as a complex matrix 2x2 of the following form,

$$1 \rightarrow \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \mathbf{i} \rightarrow \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}, \mathbf{j} \rightarrow \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \mathbf{k} \rightarrow \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$$

which then obeys the laws of multiplication as given above. An alternative definition of a quaternion is a complex matrix 2x2 of special form

$$\begin{bmatrix} q_0 + q_1 I & -q_2 - q_3 I \\ q_2 - q_3 I & q_0 - q_1 I \end{bmatrix},$$

in which q_0, q_1, q_2, q_3 are real numbers according to an above definition $q = q_0 \mathbf{1} + q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k}$. The subfield of those quaternions for which $q_2 = q_3 = 0$ is isomorphic to the field of complex numbers. If $q = w + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$, the mapping f from a quaternion to a complex matrix 2x2 is defined as

$$f(q) = \begin{bmatrix} w + Ix & y + Iz \\ -y + Iz & w - Ix \end{bmatrix},$$

equivalent to the preceding definition. A quaternionic gradient is expressed as

$$\underline{del} = \mathbf{1} \frac{1}{c} \frac{\partial}{\partial t} + \mathbf{e}_x \frac{\partial}{\partial x} + \mathbf{e}_y \frac{\partial}{\partial y} + \mathbf{e}_z \frac{\partial}{\partial z}$$

in which \mathbf{I} denotes identity, as above.

Despite a quaternion having four components, graphics in three dimensions suffice for the display of a quaternion, a quaternion curve and a quaternion surface providing that $q_0 = \sqrt{1 - q \cdot q}$ in which $q = (q_0, \mathbf{q})$ with $\mathbf{q} \cdot \mathbf{q} = 0$.

Quaternionic quantum mechanics is a modification of the complex quantum theory in which amplitude functions belong to a Hilbert space defined over the quaternionic field; quaternionic quantum mechanics appears most useful in the context of a field theory, as discussed in Adler's monograph.

The following instructions are adapted slightly from work of G. J. Fee, 2000 December, kindly provided for use here.

We define four matrices of order two,

> $cl := [One = \langle\langle 1|0\rangle, \langle 0|1\rangle\rangle, i = \langle\langle I|0\rangle, \langle 0|-I\rangle\rangle,$
 $j = \langle\langle 0|1\rangle, \langle -1|0\rangle\rangle, k = \langle\langle 0|I\rangle, \langle I|0\rangle\rangle]$;

$$cl := \left[One = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, i = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}, j = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, k = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} \right] \quad (20.1)$$

and test their multiplication property,

> $subs(cl, i).subs(cl, j) = subs(cl, k);$

$$\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} \quad (20.2)$$

> $subs(cl, j).subs(cl, k) = subs(cl, i);$

$$\begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix} \quad (20.3)$$

> $subs(cl, k).subs(cl, i) = subs(cl, j);$

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad (20.4)$$

> $subs(cl, i).subs(cl, i) = subs(cl, -One);$

(20.5)

$$\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (20.5)$$

> subs(c1,j).subs(c1,j) = subs(c1, -One);

$$\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (20.6)$$

> subs(c1,k).subs(c1,k) = subs(c1, -One);

$$\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (20.7)$$

consistent with the multiplication table above. With further substitutions we generate the complex matrix above.

> ScalarMultiply(subs(c1,One),w) + ScalarMultiply(subs(c1,i),x)
+ ScalarMultiply(subs(c1,j),y) + ScalarMultiply(subs(c1,k),z);

$$\begin{bmatrix} w + Ix & y + Iz \\ -y + Iz & w - Ix \end{bmatrix} \quad (20.8)$$

A quaternion can be viewed as a vector in a complex vector space of two dimensions, with $\mathbf{1} = (1, 0)$, $\mathbf{i} = (\sqrt{-1}, 0)$, $\mathbf{j} = (0, 1)$, $\mathbf{k} = (0, \sqrt{-1})$. We can represent a quaternion as a two-dimensional complex vector by defining

$$g(q) = (w + I^*x, y + I^*z)$$

which is just the top row of the preceding matrix.

We represent a quaternion as a real matrix 4x4 as follows,

> c4 := [One = <<1|0|0|0>, <0|1|0|0>, <0|0|1|0>, <0|0|0|1>>,
i = <<0|1|0|0>, <-1|0|0|0>, <0|0|0|-1>, <0|0|1|0>>,
j = <<0|0|1|0>, <0|0|0|1>, <-1|0|0|0>, <0|-1|0|0>>,
k = <<0|0|0|1>, <0|0|-1|0>, <0|1|0|0>, <-1|0|0|0>> 1;

$$c4 := \left[\begin{array}{l} One = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, i = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, j = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}, k \end{array} \right] \quad (20.9)$$

$$= \left[\begin{array}{l} \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix} \end{array} \right]$$

which again obeys the above laws of multiplication.

> subs(c4,i).subs(c4,j) = subs(c4, k);

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix} \quad (20.10)$$

> subs(c4,j).subs(c4,k) = subs(c4, i);

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (20.11)$$

> subs(c4,k).subs(c4,i) = subs(c4, j);

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \quad (20.12)$$

> subs(c4,i).subs(c4,i) = subs(c4, -One);

$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} = - \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (20.13)$$

> subs(c4,j).subs(c4,j) = subs(c4, -One);

$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} = - \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (20.14)$$

> subs(c4,k).subs(c4,k) = subs(c4, -One);

$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} = - \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (20.15)$$

**> ScalarMultiply(subs(c4,One),w) + ScalarMultiply(subs(c4,i),x) +
ScalarMultiply(subs(c4,j),y) + ScalarMultiply(subs(c4,k),z);**

$$\begin{bmatrix} w & x & y & z \\ -x & w & -z & y \\ -y & z & w & -x \\ -z & -y & x & w \end{bmatrix} \quad (20.16)$$

The first definition of a quaternion is then just the top row of the above matrix.

The *quaternion group* is a set of eight unit quaternions with integer components $\{+1, -1, +i, -i, +j, -j, +k, -k\}$ with respect to multiplication.

```
> gl := [1, -1, i, -i, j, -j, k, -k];
      gl := [1, -1, i, -i, j, -j, k, -k] (20.17)
```

```
> gs := [1 = Q(1,0,0,0), -1 = Q(-1,0,0,0), i = Q(0,1,0,0), -i = Q(0,
-1,0,0),
        j = Q(0,0,1,0), -j = Q(0,0,-1,0), k = Q(0,0,0,1), -k = Q(0,
0,0,-1)];
gs := [1 = Q(1,0,0,0), -1 = Q(-1,0,0,0), i = Q(0,1,0,0), -i = Q(0,-1,0,0), j = Q(0, (20.18)
0,1,0), -j = Q(0,0,-1,0), k = Q(0,0,0,1), -k = Q(0,0,0,-1)]
```

```
> gq := subs(gs, gl);
gq := [Q(1,0,0,0), Q(-1,0,0,0), Q(0,1,0,0), Q(0,-1,0,0), Q(0,0,1,0), Q(0,0,-1, (20.19)
0), Q(0,0,0,1), Q(0,0,0,-1)]
```

3.92 procedures for quaternions

```
> # no restart here
```

Four procedures that follow allow respectively addition, multiplication, subtraction and division of quaternions.

```
> `&++` := proc (a, b)
    local i;
    if nargs = 2
        then Q(op(1,a) + op(1,b), op(2,a) + op(2,b),
              op(3,a) + op(3,b), op(4,a) + op(4,b));
    elif nargs = 1 then a;
    elif nargs = 0 then Q(0,0,0,0);
    else `&+`(`&+`(a,b), seq(args[i], i=3..nargs));
    end if
end:

> `&**` := proc (a, b)
    local a1, a2, a3, a4, b1, b2, b3, b4, i;
    if nargs = 2 then a1 := op(1,a);
        a2 := op(2,a); a3 := op(3,a);
        a4 := op(4,a); b1 := op(1,b);
        b2 := op(2,b); b3 := op(3,b); b4 := op(4,b);
```

```

    Q(a1*b1 - a2*b2 - a3*b3 - a4*b4, a1*b2 + a2*b1 + a3*b4 - a4*
b3,
    a1*b3 - a2*b4 + a3*b1 + a4*b2, a1*b4 + a2*b3 - a3*b2 + a4*
b1);
    elif nargs = 1 then a;
    elif nargs = 0 then Q(1,0,0,0);
    else `&***`(`&***`(a,b), seq(args[i], i=3..nargs));
end if;
end:
> `&--` := proc (a, b)
    if nargs = 2
        then Q(op(1,a) - op(1,b), op(2,a) - op(2,b),
            op(3,a) - op(3,b), op(4,a) - op(4,b));
        elif nargs = 1 then Q(-op(1,a), -op(2,a), -op(3,a), -op(4,a));

            else ERROR("1 or 2 arguments");
        end if;
    end:
> `&//` := proc (a, b)
    local a1, a2, a3, a4, d;
    if nargs = 2 then `&***` (a, `&//`(b))
    elif nargs = 1 then a1 := op(1,a);
        a2 := op(2,a); a3 := op(3,a); a4 := op(4,a);
        d := a1^2 + a2^2 + a3^2 + a4^2;
        if d=0 then ERROR("division by null quaternion") end if;
        Q(a1/d, -a2/d, -a3/d, -a4/d);
    else ERROR("1 or 2 arguments");
    end if;
end:
> M := Matrix(8,8):
> sg := map(proc(a) rhs(a)=lhs(a) end, gs);
sg := [Q(1, 0, 0, 0)=1, Q(-1, 0, 0, 0)=-1, Q(0, 1, 0, 0)=i, Q(0, -1, 0, 0)=-i, Q(0, 0, 1, (21.1)
    0)=j, Q(0, 0, -1, 0)=-j, Q(0, 0, 0, 1)=k, Q(0, 0, 0, -1)=-k]

> for a to 8 do
    for b to 8 do
        M[a,b] := subs(sg, gq[a]&***gq[b]);
    end do;
end do;
> M;

```

$$\begin{bmatrix}
1 & -1 & i & -i & j & -j & k & -k \\
-1 & 1 & -i & i & -j & j & -k & k \\
i & -i & -1 & 1 & k & -k & -j & j \\
-i & i & 1 & -1 & -k & k & j & -j \\
j & -j & -k & k & -1 & 1 & i & -i \\
-j & j & k & -k & 1 & -1 & -i & i \\
k & -k & j & -j & -i & i & -1 & 1 \\
-k & k & -j & j & i & -i & 1 & -1
\end{bmatrix}
\quad (21.2)$$

```
> Mm := Matrix(2,2):
> Mm[1,1] := <<`*`>>;
```

$$Mm_{1,1} := \begin{bmatrix} * \end{bmatrix} \quad (21.3)$$

```
> Mm[1,2] := Matrix(gl);
```

$$Mm_{1,2} := \begin{bmatrix} 1 & -1 & i & -i & j & -j & k & -k \end{bmatrix} \quad (21.4)$$

```
> Mm[2,1] := Matrix(map(proc(a) [a] end, gl));
```

$$Mm_{2,1} := \begin{bmatrix} 1 \\ -1 \\ i \\ -i \\ j \\ -j \\ k \\ -k \end{bmatrix} \quad (21.5)$$

```
> Mm[2,2] := M;
```

$$Mm_{2,2} := \begin{bmatrix}
1 & -1 & i & -i & j & -j & k & -k \\
-1 & 1 & -i & i & -j & j & -k & k \\
i & -i & -1 & 1 & k & -k & -j & j \\
-i & i & 1 & -1 & -k & k & j & -j \\
j & -j & -k & k & -1 & 1 & i & -i \\
-j & j & k & -k & 1 & -1 & -i & i \\
k & -k & j & -j & -i & i & -1 & 1 \\
-k & k & -j & j & i & -i & 1 & -1
\end{bmatrix}
\quad (21.6)$$

This is the multiplication table for the quaternion group.

```
> `multiplication table` = Mm;
```

$$multiplication\ table = \left[\begin{array}{c} \left[\begin{array}{c} * \\ 1 \\ -1 \\ i \\ -i \\ j \\ -j \\ k \\ -k \end{array} \right] \left[\begin{array}{cccccccc} 1 & -1 & i & -i & j & -j & k & -k \end{array} \right] \\ \left[\begin{array}{cccccccc} 1 & -1 & i & -i & j & -j & k & -k \\ -1 & 1 & -i & i & -j & j & -k & k \\ i & -i & -1 & 1 & k & -k & -j & j \\ -i & i & 1 & -1 & -k & k & j & -j \\ j & -j & -k & k & -1 & 1 & i & -i \\ -j & j & k & -k & 1 & -1 & -i & i \\ k & -k & j & -j & -i & i & -1 & 1 \\ -k & k & -j & j & i & -i & 1 & -1 \end{array} \right] \end{array} \right] \quad (21.7)$$

Here is an alternative multiplication table.

```
> mM := Matrix(9):
> for i2 to 8 do for j2 to 8 do
    mM[i2+1,j2+1] := M[i2,j2] end do end do;
> mM[1,1] := Mm[1,1];
```

$$mM_{1,1} := \left[\begin{array}{c} * \end{array} \right] \quad (21.8)$$

```
> for i3 to 8 do mM[i3+1,1] := Mm[1,2][1,i3];
    mM[1,i3+1] := Mm[2,1][i3,1];
end do;
```

$$mM_{2,1} := 1$$

$$mM_{1,2} := 1$$

$$mM_{3,1} := -1$$

$$mM_{1,3} := -1$$

$$mM_{4,1} := i$$

$$mM_{1,4} := i$$

$$mM_{5,1} := -i$$

$$mM_{1,5} := -i$$

$$mM_{6,1} := j$$

$$mM_{1,6} := j$$

$$mM_{7,1} := -j$$

$$mM_{1,7} := -j$$

$$mM_{8,1} := k$$

$$\begin{aligned}
mM_{1,8} &:= k \\
mM_{9,1} &:= -k \\
mM_{1,9} &:= -k
\end{aligned}
\tag{21.9}$$

> mM;

$$\begin{bmatrix}
\begin{bmatrix} * \end{bmatrix} & 1 & -1 & i & -i & j & -j & k & -k \\
1 & 1 & -1 & i & -i & j & -j & k & -k \\
-1 & -1 & 1 & -i & i & -j & j & -k & k \\
i & i & -i & -1 & 1 & k & -k & -j & j \\
-i & -i & i & 1 & -1 & -k & k & j & -j \\
j & j & -j & -k & k & -1 & 1 & i & -i \\
-j & -j & j & k & -k & 1 & -1 & -i & i \\
k & k & -k & j & -j & -i & i & -1 & 1 \\
-k & -k & k & -j & j & i & -i & 1 & -1
\end{bmatrix}
\tag{21.10}$$

Two procedures that follow create the additive identity and the multiplicative identity. Each quaternion except the null quaternion $0 = 0 + 0 \mathbf{i} + 0 \mathbf{j} + 0 \mathbf{k}$ has an inverse.

> `&0` := proc() Q(0,0,0,0) end:

> `&1` := proc() Q(1,0,0,0) end:

This procedure forms the conjugate of a quaternion.

> Qconj := proc(a) Q(op(1,a), -op(2,a), -op(3,a), -op(4,a)) end:

Two procedures that follow form the square of the norm of a quaternion, and then the norm itself through the square root of that norm squared.

> Qnorm2 := proc(a) op(1, `&**`(a, Qconj(a))) end:

> Qnorm := proc(a) Qnorm2(a)^(1/2) end:

Two procedures that follow employ Cayley's rational parameterization of the rotation group to convert a rotation into a quaternion and vice versa.

> mat_to_q := proc(R)

local R1,R2,R3,R4,v,v1,v2,v3,nv,a,k,d,t;

R1 := 1 - R;

R2 := 1 + R;

d := LinearAlgebra[Determinant](R2);

if d=0 then

v := LinearAlgebra[LinearSolve](R1, <0,0,0>);

v := subs([_t[1]=1, _t[2]=1], v);

v1 := v[1]; v2 := v[2]; v3 := v[3];

else

R3 := 1/R2;

R4 := LinearAlgebra[Multiply](R1, R3);


```

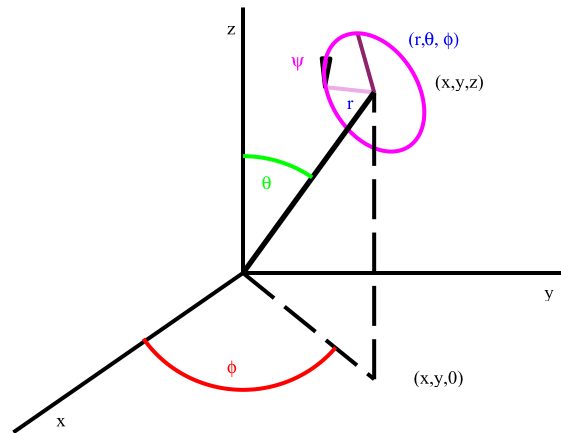
    v1 := R4[2,3];
    v2 := R4[3,1];
    v3 := R4[1,2];
end if;
nv := (v1^2 + v2^2 + v3^2)^(1/2);
v1 := v1/nv;
v2 := v2/nv;
v3 := v3/nv;
t := 1 + R[1,1] + R[2,2] + R[3,3];
a := 1/2*t^(1/2);
k := (1 - t/4)^(1/2);
v1 := k*v1;
v2 := k*v2;
v3 := k*v3;
Q(a,v1,v2,v3);
end:
> q_to_mat := proc(q)
    local q1,v1,v2,v3,v12,v13,v23,a,t,s,k,A,A1,A2,A3,A4;
    q1 := op(1,q);
    v1 := op(2,q);
    v2 := op(3,q);
    v3 := op(4,q);
    if q1=0 then
        v12 := 2*v1*v2;  v13 := 2*v1*v3;  v23 := 2*v2*v3;
        RETURN(<<-1+2*v1^2 | v12 | v13>,
                <v12 | -1+2*v2^2 | v23>,
                <v13 | v23 | -1+2*v3^2>>);
    end if;
    a := 2*arccos(q1);
    t := 1/q1^2 - 1;
    s := v1^2 + v2^2 + v3^2;
    k := (t/s)^(1/2);
    v1 := k*v1;
    v2 := k*v2;
    v3 := k*v3;
    A := <<0 | v3 | -v2>, <-v3 | 0 | v1>, <v2 | -v1 | 0>>;
    A1 := 1 - A;
    A2 := 1 + A;
    A3 := 1/A2;
    A4 := LinearAlgebra[Multiply](A1,A3);
    A4;
end:

```

Three procedures that follow provide conversion between rotational geometric parameters and quaternions, and vice versa, between rotational geometric parameters and a conventional rotation matrix. Spherical angles θ and ϕ are defined in the following diagram; ψ is the rotation angle about the axis defined by spherical angles θ and ϕ .

```
> with(plots):
  with(plottools):
    cc1 := line([0,0], [1.2,-1], linestyle=3, colour=black):
    cc2 := line([0,0], [1.2,1.7], colour=black, thickness=2):
    cc3 := point([1.2,1.7], colour=black, symbol=box):
    cc4 := line([1.2,-1], [1.2,1.7], linestyle=3, colour=black):
    cc5 := line([-2.1,-1.5], [0,0]):
    cc6 := line([0,0], [3,0]):
    cc7 := line([0,0], [0,2.5]):
    cc8 := line([1.2,1.7], [1.05,2.25], colour=maroon):
    cc9 := line([1.2,1.7], [0.75,1.75], colour=plum):
    elli := ellipse([0,0], 0.6, 0.4, colour=magenta):
    elli := rotate(elli, -Pi/3):
    elli := plottools[translate](elli, 1.2, 1.7):
    cc10 := elli:
    cc11 := arrow([0.76,2.02], [0.75,1.75], .02, .08, 1, colour=
magenta):
    ct1 := textplot([-0.1,-0.9,"f"], colour=red, font=[SYMBOL]):
    ct2 := textplot([-1.65,-1.4,"x"], colour=black):
    ct3 := textplot([2.8,-0.2,"y"], colour=black):
    ct4 := textplot([-0.1,2.3,"z"], colour=black):
    ct5 := textplot([0.21,0.85,"q"], colour=green, font = [SYMBOL]):
    ct6 := textplot([1.0,1.6,"r"], colour=blue):
    ct7 := plot([1.1*cos(t),1.1*sin(t), t=Pi*0.305..Pi/2], colour=
green):
    ct8 := plot([1.1*cos(t),1.1*sin(t), t=Pi*1.19..Pi*1.78], colour=
red):
    ct9 := textplot([2.0,1.8,"(x,y,z)"]):
    ct10 := textplot([1.8,-1.0,"(x,y,0)"]):
    ct11 := textplot([1.60,2.2,"(r,"], colour=blue):
    ct12 := textplot([1.82,2.2,"q, f)"], colour=blue, font=[SYMBOL]):
    ct13 := textplot([0.5,2.0,"y"], colour=magenta, font=[SYMBOL]):
    display({seq(cc||i, i=1..11), seq(ct||i, i=1..13)},
            title ="definition of rotational angles", view=[-3..3,-3.
.3],
            tickmarks=[2,2], labels=[y,z], axes=none,
            titlefont=[TIMES,BOLD,14]);
```

definition of rotational angles



In this diagram, θ denotes the polar angle between polar axis z and a vector from the origin to point with coordinates (x,y,z) or (r,θ,ϕ) ; ϕ denotes the equatorial angle between the projection of the vector in equatorial plane xy or $z = 0$ and axis x , according to a conventional right-hand rule.

```
> gp_to_q := proc(pr::PHYS_ROT(algebraic,algebraic,algebraic))
    local x,y,z,c2,s2,theta,phi,psi;
    description "theta and phi are spherical coordinates that define
the axis of rotation",
    "theta is the angle in radians from the positive z-axis",
    "phi is the angle from the positive x-axis to the projection
onto the x-y plane",
    "psi is the rotation angle, using the right-hand rule",
    "the output is the unit quaternion that represents the above
physical rotation";
    theta := op(1, pr);
    phi := op(2, pr);
    psi := op(3, pr);
    x := sin(theta)*cos(phi);
    y := sin(theta)*sin(phi);
    z := cos(theta);
    c2 := cos(psi/2);
    s2 := sin(psi/2);
    Q(c2,s2*x,s2*y,s2*z);
end:

> gp_to_mat := proc(pr::PHYS_ROT(algebraic,algebraic,algebraic))
    local theta,phi,psi,ct,st,ch,sh,cs,ss,M1,M1i,M2,M2i,M3,M;
```

```

description "theta and phi are spherical coordinates that define
the axis of rotation",
  "theta is the angle in radians from the positive z-axis",
  "phi is the angle from the positive x-axis to the projection
onto the x-y plane",
  "psi is the rotation angle, using the right-hand rule",
  "the output is the unit quaternion that represents the above
physical rotation";
theta := op(1,pr);
phi := op(2,pr);
psi := op(3,pr);
ct := cos(theta); st := sin(theta);
ch := cos(phi); sh := sin(phi);
cs := cos(psi); ss := sin(psi);
M1 := <<ch|-sh|0>,<sh|ch|0>,<0|0|1>>;
M1i := <<ch|sh|0>,<-sh|ch|0>,<0|0|1>>;
M2 := <<ct|0|st>,<0|1|0>,<-st|0|ct>>;
M2i := <<ct|0|-st>,<0|1|0>,<st|0|ct>>;
M3 := <<cs|-ss|0>,<ss|cs|0>,<0|0|1>>;
M := LinearAlgebra[Multiply](M2i,M1i);
M := LinearAlgebra[Multiply](M3,M);
M := LinearAlgebra[Multiply](M2,M);
M := LinearAlgebra[Multiply](M1,M);
M;
end:
> q_to_gp := proc(q::Q(algebraic,algebraic,algebraic,algebraic))
  local x, y, z, c2, psi, phi, s2, theta;
  description " inverse of above";
  x := op(2,q);
  y := op(3,q);
  z := op(4,q);
  c2 := op(1,q);
  psi := 2*arccos(c2);
  phi := arctan(y,x);
  s2 := (1-c2^2)^(1/2);
  if s2=0 then theta := 0; else theta := arccos(z/s2); fi;
  PHYS_ROT(theta,phi,psi);
end:
> gp_to_mat2 := proc(P)
  q_to_mat(gp_to_q(P));
end:

```

This procedure converts a rotational matrix to physical parameters.

```
> mat_to_gp := proc(R)
    q_to_gp(mat_to_q(R));
end:
```

This procedure combines two rotations given in physical parameters into the composite rotation also given in physical parameters.

```
> gp_mul := proc(A,B)
    local q1, q2, q3;
    q1 := gp_to_q(A);
    q2 := gp_to_q(B);
    q3 := q1 &** q2;
    q_to_gp(q3);
end:
```

This procedure rotates a vector with a given rotation matrix.

```
> mat_v := proc(R,v) LinearAlgebra[Multiply](R,v) end:
```

This procedure rotates a vector given in quaternion form by a rotation given in quaternion form.

```
> q_v := proc(q, v)
    q &** v &** Qconj(q);
end:
```

This procedure rotates a vector by a rotation given in physical parameters.

```
> gp_v := proc(P,v)
    LinearAlgebra[Multiply](gp_to_mat(P),v);
end:
```

This procedure tests if input is a rotation matrix.

```
> is_mat := proc(R)
    local m,n,Rt,RRt,eps,i,j,d;
    if type(R, Matrix) then
        m := LinearAlgebra[RowDimension](R);
        n := LinearAlgebra[ColumnDimension](R);
        if m=3 and n=3 then
            Rt := LinearAlgebra[Transpose](R);
            RRt := LinearAlgebra[Multiply](R,Rt);
            RRt := RRt - 1;
            RRt := map(simplify, RRt);
            RRt := map(combine, RRt);
            RRt := map(evalf, RRt);
            eps := Float(1, 4-Digits);
            for i to 3 do
                for j to 3 do
                    if abs(RRt[i,j])>eps then RETURN(false); end if;
                end do;
            end do;
            d := LinearAlgebra[Determinant](R);
```

```

    d := simplify(d);
    d := combine(d);
    d := evalf(d);
    d := d - 1;
    if abs(d)>eps then RETURN(false); end if;
    RETURN(true);
end if;
end if;
false;
end:

```

These two procedures convert eulerian angles to a rotational matrix and vice versa.

```

> ea_to_mat := proc(A::EA(algebraic,algebraic,algebraic),
                    l::[integer,integer,integer])
    local a,b,c,d,e, phi,theta,psi;
    psi := op(1,A);
    theta := op(2,A);
    phi := op(3,A);
    b := <<cos(psi) | sin(psi) | 0>,
        <-sin(psi) | cos(psi) | 0>,
        <0 | 0 | 1>>;
    c := <<1 | 0 | 0>,
        <0 | cos(theta) | sin(theta)>,
        <0 | -sin(theta) | cos(theta)>>;
    d := <<cos(phi) | sin(phi) | 0>,
        <-sin(phi) | cos(phi) | 0>,
        <0 | 0 | 1>>;
    e := <<cos(theta) | 0 | -sin(theta)>,
        <0 | 1 | 0>,
        <sin(theta) | 0 | cos(theta)>>;
    if l[1]=3 and l[2]=1 and l[3]=3
        then a := b . c . d
        elif l[1]=3 and l[2]=2 and l[3]=3
            then a := b . e . d
            elif l[1]=1 and l[2]=2 and l[3]=3
                then a := subs(theta=phi, c) . e . b;
        end if;
    a;
end:

> mat_to_ea := proc(M::Matrix, l::[integer,integer,integer])
    local st,phi,theta,psi;
    if l[1]=3 and l[2]=1 and l[3]=3 then
        theta := arccos(M[3,3]);

```

```

    st := sin(theta);
#   phi := arccos(-M[3,2]/st);
#   psi := arccos(M[2,3]/st);
    if st=0 then
        psi := 0;
        phi := arctan(M[1,2], M[1,1]);
    else
        phi := arctan(M[3,1]/st, - M[3,2]/st);
        psi := arctan(M[1,3]/st, M[2,3]/st);
    end if;
    EA(psi,theta,phi);
end if;
end:

```

3.93 applications and tests of quaternions

> # no restart here

As an application, quaternion multiplication can be used as an aid to solve the problem about the sum of four squares. This problem is stated as

$$\begin{aligned}
 > \text{eq} := w^2 + x^2 + y^2 + z^2 = n; \\
 &\text{eq} := w^2 + x^2 + y^2 + z^2 = n
 \end{aligned}
 \tag{22.1}$$

in which n is a given integer and the objective is to find integers w, x, y, z . Here follow some examples. For $n = 23$

$$\begin{aligned}
 > \text{eq1} := x1^2 + x2^2 + x3^2 + x4^4 = 23; \\
 &\text{eq1} := x4^4 + x1^2 + x2^2 + x3^2 = 23
 \end{aligned}
 \tag{22.2}$$

a solution is

$$\begin{aligned}
 > \text{ss1} := [x1=3, x2=3, x3=2, x4=1]; \\
 &\text{ss1} := [x1=3, x2=3, x3=2, x4=1]
 \end{aligned}
 \tag{22.3}$$

$$\begin{aligned}
 > \text{subs}(\text{ss1}, \text{eq1}); \\
 &23 = 23
 \end{aligned}
 \tag{22.4}$$

For $n = 31$,

$$\begin{aligned}
 > \text{eq2} := y1^2 + y2^2 + y3^2 + y4^2 = 31; \\
 &\text{eq2} := y1^2 + y2^2 + y3^2 + y4^2 = 31
 \end{aligned}
 \tag{22.5}$$

a solution is

$$\begin{aligned}
 > \text{ss2} := [y1=5, y2=2, y3=1, y4=1]; \\
 &\text{ss2} := [y1=5, y2=2, y3=1, y4=1]
 \end{aligned}
 \tag{22.6}$$

$$\begin{aligned}
 > \text{subs}(\text{ss2}, \text{eq2}); \\
 &31 = 31
 \end{aligned}
 \tag{22.7}$$

For an integer being the product of two preceding values of n ,

$$\begin{aligned} > \text{eq3} := z1^2 + z2^2 + z3^2 + z4^2 = \text{rhs}(\text{eq1}) * \text{rhs}(\text{eq2}); \\ &eq3 := z1^2 + z2^2 + z3^2 + z4^2 = 713 \end{aligned} \quad (22.8)$$

a solution is obtained as follows using quaternions.

$$\begin{aligned} > q1 := \text{subs}(\text{ss1}, Q(x1, x2, x3, x4)); \\ &q1 := Q(3, 3, 2, 1) \end{aligned} \quad (22.9)$$

$$\begin{aligned} > q2 := \text{subs}(\text{ss2}, Q(y1, y2, y3, y4)); \\ &q2 := Q(5, 2, 1, 1) \end{aligned} \quad (22.10)$$

$$\begin{aligned} > q3 := q1 \&** q2; \\ &q3 := Q(6, 22, 12, 7) \end{aligned} \quad (22.11)$$

$$\begin{aligned} > \text{ss3} := [z1=\text{op}(1, q3), z2=\text{op}(2, q3), z3=\text{op}(3, q3), z4=\text{op}(4, q3)]; \\ &\text{ss3} := [z1=6, z2=22, z3=12, z4=7] \end{aligned} \quad (22.12)$$

$$\begin{aligned} > \text{subs}(\text{ss3}, \text{eq3}); \\ &713 = 713 \end{aligned} \quad (22.13)$$

Other tests and illustrations follow.

$$\begin{aligned} > \text{\&0}(); \\ &Q(0, 0, 0, 0) \end{aligned} \quad (22.14)$$

$$\begin{aligned} > \text{\&1}(); \\ &Q(1, 0, 0, 0) \end{aligned} \quad (22.15)$$

$$\begin{aligned} > q1 := Q(2, 1, 1, 1); \\ &q1 := Q(2, 1, 1, 1) \end{aligned} \quad (22.16)$$

$$\begin{aligned} > q2 := Q(1, 1, 1, 0); \\ &q2 := Q(1, 1, 1, 0) \end{aligned} \quad (22.17)$$

$$\begin{aligned} > Q\text{norm}(q2); \\ &\sqrt{3} \end{aligned} \quad (22.18)$$

$$\begin{aligned} > q3 := q1 \&** q2; \\ &q3 := Q(0, 2, 4, 1) \end{aligned} \quad (22.19)$$

$$\begin{aligned} > Q(2, 2, 2, 0) \&++ Q(-1, 1, 0, 1) \&++ Q(-1, 0, 1, -1) \&++ Q(0, -1, 1, 1); \\ &Q(0, 2, 4, 1) \end{aligned} \quad (22.20)$$

$$\begin{aligned} > Q\text{conj}(q1); \\ &Q(2, -1, -1, -1) \end{aligned} \quad (22.21)$$

$$\begin{aligned} > Q\text{norm}(q1) * Q\text{norm}(q2) = Q\text{norm}(q1 \&** q2); \\ &\sqrt{7} \sqrt{3} = \sqrt{21} \end{aligned} \quad (22.22)$$

To show that each quaternion except &0 has an inverse we perform the following operations.

$$\begin{aligned} > q4 := Q(a1, a2, a3, a4); \\ &q4 := Q(a1, a2, a3, a4) \end{aligned} \quad (22.23)$$

$$\begin{aligned} &> \text{q5} := Q(b1, b2, b3, b4); \\ &q5 := Q(b1, b2, b3, b4) \end{aligned} \quad (22.24)$$

$$\begin{aligned} &> \text{q6} := \text{'\&1'()}; \\ &q6 := Q(1, 0, 0, 0) \end{aligned} \quad (22.25)$$

$$\begin{aligned} &> \text{q45} := \text{q4} \& \text{q5}; \\ q45 &:= Q(a1\,b1 - a2\,b2 - a3\,b3 - a4\,b4, a1\,b2 + b1\,a2 + a3\,b4 - a4\,b3, a1\,b3 - a2\,b4 \\ &\quad + a3\,b1 + a4\,b2, a1\,b4 + a2\,b3 - a3\,b2 + a4\,b1) \end{aligned} \quad (22.26)$$

$$\begin{aligned} &> \text{for ii to 4 do eq[ii] := op(ii, q45) = op(ii, q6) end do;} \\ eq_1 &:= a1\,b1 - a2\,b2 - a3\,b3 - a4\,b4 = 1 \\ eq_2 &:= a1\,b2 + b1\,a2 + a3\,b4 - a4\,b3 = 0 \\ eq_3 &:= a1\,b3 - a2\,b4 + a3\,b1 + a4\,b2 = 0 \\ eq_4 &:= a1\,b4 + a2\,b3 - a3\,b2 + a4\,b1 = 0 \end{aligned} \quad (22.27)$$

$$\begin{aligned} &> \text{eqs} := \text{convert(eq, set)}; \\ eqs &:= \{a1\,b4 + a2\,b3 - a3\,b2 + a4\,b1 = 0, a1\,b3 - a2\,b4 + a3\,b1 + a4\,b2 = 0, a1\,b2 \\ &\quad + b1\,a2 + a3\,b4 - a4\,b3 = 0, a1\,b1 - a2\,b2 - a3\,b3 - a4\,b4 = 1\} \end{aligned} \quad (22.28)$$

$$\begin{aligned} &> \text{ss} := \text{solve(eqs, \{a1, a2, a3, a4\})}; \\ ss &:= \left\{ a1 = \frac{b1}{b1^2 + b2^2 + b3^2 + b4^2}, a2 = -\frac{b2}{b1^2 + b2^2 + b3^2 + b4^2}, a3 = \right. \\ &\quad \left. -\frac{b3}{b1^2 + b2^2 + b3^2 + b4^2}, a4 = -\frac{b4}{b1^2 + b2^2 + b3^2 + b4^2} \right\} \end{aligned} \quad (22.29)$$

The only case in which this solution is undefined would be for $\{b1=0, b2=0, b3=0, b4=0\}$.

$$\begin{aligned} &> \text{r1} := \text{PHYS_ROT}(0.12, 0.13, 0.14); \\ r1 &:= \text{PHYS_ROT}(0.12, 0.13, 0.14) \end{aligned} \quad (22.30)$$

$$\begin{aligned} &> \text{q1} := \text{gp_to_q}(r1); \\ q1 &:= Q(0.9975510003, 0.008302360271, 0.001085428314, 0.06943986286) \end{aligned} \quad (22.31)$$

$$\begin{aligned} &> \text{q_to_gp}(q1); \\ &\text{PHYS_ROT}(0.1199999203, 0.1299999999, 0.1399999987) \end{aligned} \quad (22.32)$$

$$\begin{aligned} &> \text{Qnorm}(q1); \\ &1.000000000 \end{aligned} \quad (22.33)$$

$$\begin{aligned} &> \text{r2} := \text{PHYS_ROT}(0.21, 0.23, 0.09); \\ r2 &:= \text{PHYS_ROT}(0.21, 0.23, 0.09) \end{aligned} \quad (22.34)$$

$$\begin{aligned} &> \text{q2} := \text{gp_to_q}(r2); \\ q2 &:= Q(0.9989876708, 0.009130585659, 0.002137866027, 0.04399653882) \end{aligned} \quad (22.35)$$

$$> \text{q_to_gp}(q2);$$

...

$$PHYS_ROT(0.2100001234, 0.23000000001, 0.090000000212) \quad (22.36)$$

$$> Qnorm(q2); \quad 0.9999999999 \quad (22.37)$$

$$> r3 := gp_mul(r1, r2); \quad r3 := PHYS_ROT(0.1545768030, 0.1988076069, 0.2297710960) \quad (22.38)$$

$$> q3 := q1 \&** q2; \quad q3 := Q(0.9934079108, 0.01730148238, 0.003485711397, 0.1132661969) \quad (22.39)$$

$$> q_to_gp(q3); \quad PHYS_ROT(0.1545768030, 0.1988076069, 0.2297710960) \quad (22.40)$$

$$> Qnorm(q3); \quad 1.0000000000 \quad (22.41)$$

$$> r_q3 := q_to_gp(q3); \quad r_q3 := PHYS_ROT(0.1545768030, 0.1988076069, 0.2297710960) \quad (22.42)$$

$$> pr := q_to_gp(q3); \quad pr := PHYS_ROT(0.1545768030, 0.1988076069, 0.2297710960) \quad (22.43)$$

$$> r1 := PHYS_ROT(0, 0, Pi/2); \quad r1 := PHYS_ROT\left(0, 0, \frac{\pi}{2}\right) \quad (22.44)$$

$$> r2 := PHYS_ROT(Pi/2, Pi/2, Pi/2); \quad r2 := PHYS_ROT\left(\frac{\pi}{2}, \frac{\pi}{2}, \frac{\pi}{2}\right) \quad (22.45)$$

$$> r3 := PHYS_ROT(arccos(1/sqrt(3)), Pi/4, 2*Pi/3); \quad r3 := PHYS_ROT\left(\arccos\left(\frac{\sqrt{3}}{3}\right), \frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.46)$$

$$> r1_2 := map(evalf, gp_mul(r1, r2)); \quad r1_2 := PHYS_ROT(0.9553166179, 2.356194490, 2.094395103) \quad (22.47)$$

$$> gp_to_q(r1_2); \quad Q(0.4999999993, -0.50000000001, 0.50000000002, 0.50000000004) \quad (22.48)$$

$$> q_to_gp(%); \quad PHYS_ROT(0.9553166179, 2.356194490, 2.094395104) \quad (22.49)$$

$$> r4 := PHYS_ROT(Pi/2, Pi/2, -Pi/2); \quad r4 := PHYS_ROT\left(\frac{\pi}{2}, \frac{\pi}{2}, -\frac{\pi}{2}\right) \quad (22.50)$$

$$> r5 := PHYS_ROT(Pi/2, 0, Pi/2); \quad (22.51)$$

$$r5 := PHYS_ROT\left(\frac{\pi}{2}, 0, \frac{\pi}{2}\right) \quad (22.51)$$

> r6 := PHYS_ROT(Pi-arccos(1/sqrt(3)), -Pi/4, 2*Pi/3);

$$r6 := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}}{3}\right), -\frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.52)$$

> r4_5 := gp_mul(r4, r5);

$$r4_5 := PHYS_ROT\left(\arccos\left(\frac{\sqrt{3}\sqrt{4}}{6}\right), -\frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.53)$$

> r4_5 := map(simplify, r4_5);

$$r4_5 := PHYS_ROT\left(\arccos\left(\frac{\sqrt{3}}{3}\right), -\frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.54)$$

> r7 := PHYS_ROT(Pi/2, Pi/2, Pi/2);

$$r7 := PHYS_ROT\left(\frac{\pi}{2}, \frac{\pi}{2}, \frac{\pi}{2}\right) \quad (22.55)$$

> r8 := PHYS_ROT(0, 0, Pi/2);

$$r8 := PHYS_ROT\left(0, 0, \frac{\pi}{2}\right) \quad (22.56)$$

> r9 := PHYS_ROT(arccos(1/sqrt(3)), 3*Pi/4, 2*Pi/3);

$$r9 := PHYS_ROT\left(\arccos\left(\frac{\sqrt{3}}{3}\right), \frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.57)$$

> r7_8 := gp_mul(r7, r8);

$$r7_8 := PHYS_ROT\left(\arccos\left(\frac{\sqrt{3}\sqrt{4}}{6}\right), \frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.58)$$

> r7_8 := map(simplify, r7_8);

$$r7_8 := PHYS_ROT\left(\arccos\left(\frac{\sqrt{3}}{3}\right), \frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.59)$$

> q20 := Q(1, 2, 3, 4);

$$q20 := Q(1, 2, 3, 4) \quad (22.60)$$

> nQ20 := evalf(Qnorm(q20));

$$nQ20 := 5.477225575 \quad (22.61)$$

> q20 := q20 & // Q(nQ20, 0, 0, 0);

$$q20 := Q(0.1825741858, 0.3651483716, 0.5477225574, 0.7302967432) \quad (22.62)$$

> m20 := q_to_mat(q20);

$$(22.63)$$

$$m20 := \begin{bmatrix} -0.666666666755556 & 0.133333333231111 & 0.733333333271111 \\ 0.666666666555556 & -0.333333333511111 & 0.666666666688889 \\ 0.333333333377778 & 0.933333333284444 & 0.1333333333564444 \end{bmatrix} \quad (22.63)$$

$$> \text{is_mat}(m20); \quad \text{true} \quad (22.64)$$

$$> q30 := \&-- q20; \\ q30 := Q(-0.1825741858, -0.3651483716, -0.5477225574, -0.7302967432) \quad (22.65)$$

$$> m30 := q_to_mat(q30); \\ m30 := \begin{bmatrix} -0.666666666755556 & 0.666666666555556 & 0.333333333377778 \\ 0.133333333231111 & -0.333333333511111 & 0.933333333284444 \\ 0.733333333271111 & 0.666666666688889 & 0.1333333333564444 \end{bmatrix} \quad (22.66)$$

$$> r20 := q_to_gp(q20); \\ r20 := \text{PHYS_ROT}(0.7335813240, 0.9827937232, 2.774384634) \quad (22.67)$$

$$> r30 := q_to_gp(q30); \\ r30 := \text{PHYS_ROT}(2.408011330, -2.158798930, 3.508800674) \quad (22.68)$$

$$> R20 := gp_to_mat(r20); \\ > R20 := \text{evalf}(\%); \\ R20 := \begin{bmatrix} -0.666666666722017 & 0.1333333334525506 & 0.733333333138296 \\ 0.666666666320149 & -0.333333333192568 & 0.6666666667176528 \\ 0.3333333334034305 & 0.933333333293929 & 0.1333333332499701 \end{bmatrix} \quad (22.69)$$

$$> R30 := gp_to_mat(r30); \\ > \text{evalf}(\%); \\ \begin{bmatrix} -0.666666666953376 & 0.133333333321302 & 0.7333333332979705 \\ 0.666666666388831 & -0.3333333333063778 & 0.666666666960413 \\ 0.3333333333067458 & 0.933333333360257 & 0.133333333306236 \end{bmatrix} \quad (22.70)$$

$$> \arccos(-0.5); \quad 2.094395102 \quad (22.71)$$

$$> p1 := \text{PHYS_ROT}(\arccos(1/\sqrt{3}), \text{Pi}/4, 2*\text{Pi}/3); \\ p1 := \text{PHYS_ROT}\left(\arccos\left(\frac{\sqrt{3}}{3}\right), \frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.72)$$

$$> p1 := \text{PHYS_ROT}(-2*\arccos(1/\sqrt{3}), -1*\text{Pi}/4, -1*\text{Pi}/6); \\ p1 := \text{PHYS_ROT}\left(-2 \arccos\left(\frac{\sqrt{3}}{3}\right), -\frac{\pi}{4}, -\frac{\pi}{6}\right) \quad (22.73)$$

> r1 := gp_to_mat(p1):

> r1 := map(simplify, r1);

$$r1 := \begin{bmatrix} \left[\frac{\left((-2 + \sqrt{3}) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)\right)^2}{4} + \frac{\sqrt{3}}{4} + \frac{1}{2}, \right. \\ \frac{\left((2 - \sqrt{3}) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)\right)^2}{4} + \frac{\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)}{2} + \frac{\sqrt{3}}{4} - \frac{1}{2}, \\ \left. \frac{\sqrt{2} \left(-1 + (-2 + \sqrt{3}) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)\right) \sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)}{4} \right] \\ \left[\frac{\left((2 - \sqrt{3}) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)\right)^2}{4} - \frac{\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)}{2} + \frac{\sqrt{3}}{4} - \frac{1}{2}, \right. \\ \frac{\left((-2 + \sqrt{3}) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)\right)^2}{4} + \frac{\sqrt{3}}{4} + \frac{1}{2}, \\ \left. - \frac{\sqrt{2} \sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \left(1 + (-2 + \sqrt{3}) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \right)}{4} \right] \\ \left[\frac{\sqrt{2} \sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \left(1 + (-2 + \sqrt{3}) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \right)}{4}, \right. \\ \left. - \frac{\sqrt{2} \left(-1 + (-2 + \sqrt{3}) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)\right) \sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)}{4}, \right. \\ \left. \left. \frac{\left((2 - \sqrt{3}) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)\right)^2}{2} + \frac{\sqrt{3}}{2} \right] \right] \end{bmatrix}, \quad (22.74)$$

> r2 := gp_to_mat2(p1):

> r2 := map(simplify, r2);

$$r2 := \left[\left[\frac{-\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)^2 + 4\sqrt{3} + 7}{4\sqrt{3} + 8}, \right. \right. \quad (22.75)$$

$$\begin{aligned}
& \frac{-1 + \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)^2 + (2\sqrt{3} + 4) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)}{4\sqrt{3} + 8}, \\
& - \frac{\sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sqrt{2} \left(\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) + 2 + \sqrt{3}\right)}{4\sqrt{3} + 8} \Bigg], \\
& \left[\frac{-1 + \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)^2 + (-2\sqrt{3} - 4) \cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)}{4\sqrt{3} + 8}, \right. \\
& \frac{-\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)^2 + 4\sqrt{3} + 7}{4\sqrt{3} + 8}, \\
& \left. - \frac{\sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sqrt{2} \left(-\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) + 2 + \sqrt{3}\right)}{4\sqrt{3} + 8} \right], \\
& \left[\frac{\sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sqrt{2} \left(-\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) + 2 + \sqrt{3}\right)}{4\sqrt{3} + 8}, \right. \\
& \frac{\sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sqrt{2} \left(\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) + 2 + \sqrt{3}\right)}{4\sqrt{3} + 8}, \\
& \left. \frac{\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right)^2 + 2\sqrt{3} + 3}{2\sqrt{3} + 4} \right] \Bigg]
\end{aligned}$$

> map(expand, r1);

$$\begin{aligned}
& \left[\left[\frac{4}{9} + \frac{5\sqrt{3}}{18}, -\frac{11}{18} + \frac{2\sqrt{3}}{9}, -\frac{\sqrt{2}\sqrt{6}\sqrt{3}}{54} - \frac{\sqrt{2}\sqrt{6}}{18} \right], \right. \\
& \left[-\frac{5}{18} + \frac{2\sqrt{3}}{9}, \frac{4}{9} + \frac{5\sqrt{3}}{18}, -\frac{5\sqrt{2}\sqrt{6}\sqrt{3}}{54} + \frac{\sqrt{2}\sqrt{6}}{18} \right], \\
& \left[\frac{5\sqrt{2}\sqrt{6}\sqrt{3}}{54} - \frac{\sqrt{2}\sqrt{6}}{18}, \frac{\sqrt{2}\sqrt{6}\sqrt{3}}{54} + \frac{\sqrt{2}\sqrt{6}}{18}, \frac{1}{9} + \frac{4\sqrt{3}}{9} \right] \Bigg]
\end{aligned} \tag{22.76}$$

> map(expand, r2);

$$\begin{aligned}
& \left[\left[\frac{62}{9(4\sqrt{3}+8)} + \frac{4\sqrt{3}}{4\sqrt{3}+8}, -\frac{20}{9(4\sqrt{3}+8)} - \frac{2\sqrt{3}}{3(4\sqrt{3}+8)}, -\frac{10\sqrt{2}\sqrt{6}\sqrt{3}}{27(4\sqrt{3}+8)} \right. \right. \\
& \quad \left. \left. - \frac{2\sqrt{2}\sqrt{6}}{3(4\sqrt{3}+8)} \right], \right. \\
& \left[\frac{4}{9(4\sqrt{3}+8)} + \frac{2\sqrt{3}}{3(4\sqrt{3}+8)}, \frac{62}{9(4\sqrt{3}+8)} + \frac{4\sqrt{3}}{4\sqrt{3}+8}, \right. \\
& \quad \left. -\frac{14\sqrt{2}\sqrt{6}\sqrt{3}}{27(4\sqrt{3}+8)} - \frac{2\sqrt{2}\sqrt{6}}{3(4\sqrt{3}+8)} \right], \\
& \left[\frac{14\sqrt{2}\sqrt{6}\sqrt{3}}{27(4\sqrt{3}+8)} + \frac{2\sqrt{2}\sqrt{6}}{3(4\sqrt{3}+8)}, \frac{10\sqrt{2}\sqrt{6}\sqrt{3}}{27(4\sqrt{3}+8)} + \frac{2\sqrt{2}\sqrt{6}}{3(4\sqrt{3}+8)}, \right. \\
& \quad \left. \frac{28}{9(2\sqrt{3}+4)} + \frac{2\sqrt{3}}{2\sqrt{3}+4} \right] \Big]
\end{aligned} \tag{22.77}$$

> map(normal,%,expanded);

$$\begin{aligned}
& \left[\left[\frac{31+18\sqrt{3}}{18\sqrt{3}+36}, \frac{-10-3\sqrt{3}}{18\sqrt{3}+36}, \frac{-5\sqrt{2}\sqrt{6}\sqrt{3}-9\sqrt{2}\sqrt{6}}{54\sqrt{3}+108} \right], \right. \\
& \left[\frac{2+3\sqrt{3}}{18\sqrt{3}+36}, \frac{31+18\sqrt{3}}{18\sqrt{3}+36}, \frac{-7\sqrt{2}\sqrt{6}\sqrt{3}-9\sqrt{2}\sqrt{6}}{54\sqrt{3}+108} \right], \\
& \left[\frac{7\sqrt{2}\sqrt{6}\sqrt{3}+9\sqrt{2}\sqrt{6}}{54\sqrt{3}+108}, \frac{5\sqrt{2}\sqrt{6}\sqrt{3}+9\sqrt{2}\sqrt{6}}{54\sqrt{3}+108}, \frac{14+9\sqrt{3}}{9\sqrt{3}+18} \right] \Big]
\end{aligned} \tag{22.78}$$

> map(simplify,%);

$$\begin{aligned}
& \left[\begin{array}{ccc} \frac{31+18\sqrt{3}}{18\sqrt{3}+36} & \frac{-10-3\sqrt{3}}{18\sqrt{3}+36} & \frac{-5-3\sqrt{3}}{9\sqrt{3}+18} \\ \frac{2+3\sqrt{3}}{18\sqrt{3}+36} & \frac{31+18\sqrt{3}}{18\sqrt{3}+36} & \frac{-7-3\sqrt{3}}{9\sqrt{3}+18} \\ \frac{7+3\sqrt{3}}{9\sqrt{3}+18} & \frac{5+3\sqrt{3}}{9\sqrt{3}+18} & \frac{14+9\sqrt{3}}{9\sqrt{3}+18} \end{array} \right]
\end{aligned} \tag{22.79}$$

> map(normal,%,expanded);

(22.80)

$$\begin{bmatrix} \frac{31 + 18\sqrt{3}}{18\sqrt{3} + 36} & \frac{-10 - 3\sqrt{3}}{18\sqrt{3} + 36} & \frac{-5 - 3\sqrt{3}}{9\sqrt{3} + 18} \\ \frac{2 + 3\sqrt{3}}{18\sqrt{3} + 36} & \frac{31 + 18\sqrt{3}}{18\sqrt{3} + 36} & \frac{-7 - 3\sqrt{3}}{9\sqrt{3} + 18} \\ \frac{7 + 3\sqrt{3}}{9\sqrt{3} + 18} & \frac{5 + 3\sqrt{3}}{9\sqrt{3} + 18} & \frac{14 + 9\sqrt{3}}{9\sqrt{3} + 18} \end{bmatrix} \quad (22.80)$$

> gp_to_q(p1);

$$Q\left(\cos\left(\frac{\pi}{12}\right), \frac{\sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sqrt{2} \sin\left(\frac{\pi}{12}\right)}{2}, \right. \\ \left. - \frac{\sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sqrt{2} \sin\left(\frac{\pi}{12}\right)}{2}, -\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sin\left(\frac{\pi}{12}\right)\right) \quad (22.81)$$

> map(evalf,%);

$$Q(0.9659258263, 0.1725460300, -0.1725460300, 0.08627301497) \quad (22.82)$$

> map(expand,%);

$$Q\left(\cos\left(\frac{\pi}{12}\right), \frac{\sqrt{2} \sin\left(\frac{\pi}{12}\right) \sqrt{6} \sqrt{3}}{9}, -\frac{\sqrt{2} \sin\left(\frac{\pi}{12}\right) \sqrt{6} \sqrt{3}}{9}, \frac{\sin\left(\frac{\pi}{12}\right)}{3}\right) \quad (22.83)$$

> map(simplify,%);

$$Q\left(\cos\left(\frac{\pi}{12}\right), \frac{2 \sin\left(\frac{\pi}{12}\right)}{3}, -\frac{2 \sin\left(\frac{\pi}{12}\right)}{3}, \frac{\sin\left(\frac{\pi}{12}\right)}{3}\right) \quad (22.84)$$

> M1 := gp_to_mat(PHYS_ROT(a,b,c));

$$M1 := \begin{bmatrix} [\cos(9) (\cos(9) (\cos(c) \cos(9)^2 + \sin(c) \sin(9)) + \cos(9) \sin(9)^2) \\ - \sin(9) (\sin(c) \cos(9)^2 - \cos(c) \sin(9)), \cos(9) (\cos(9) (\cos(c) \cos(9) \sin(9) \\ - \sin(c) \cos(9)) + \sin(9)^3) - \sin(9) (\sin(c) \cos(9) \sin(9) + \cos(c) \cos(9)), \cos(9) (\\ - \cos(c) \cos(9) \sin(9) + \cos(9) \sin(9)) + \sin(9)^2 \sin(c)], \\ [\sin(9) (\cos(9) (\cos(c) \cos(9)^2 + \sin(c) \sin(9)) + \cos(9) \sin(9)^2) \\ + \cos(9) (\sin(c) \cos(9)^2 - \cos(c) \sin(9)), \sin(9) (\cos(9) (\cos(c) \cos(9) \sin(9) \\ - \sin(c) \cos(9)) + \sin(9)^3) + \cos(9) (\sin(c) \cos(9) \sin(9) + \cos(c) \cos(9)), \sin(9) (\\ - \cos(c) \cos(9) \sin(9) + \cos(9) \sin(9)) - \sin(c) \cos(9) \sin(9)], \\ [-\sin(9) (\cos(c) \cos(9)^2 + \sin(c) \sin(9)) + \cos(9)^2 \sin(9), \end{bmatrix} \quad (22.85)$$

$$-\sin(9) (\cos(c) \cos(9) \sin(9) - \sin(c) \cos(9)) + \cos(9) \sin(9)^2, \sin(9)^2 \cos(c) + \cos(9)^2]]$$

> map(length, M1);

$$\begin{bmatrix} 201 & 213 & 117 \\ 201 & 213 & 129 \\ 105 & 117 & 45 \end{bmatrix} \quad (22.86)$$

> MM := map(simplify, M1);

$$\begin{aligned} MM := & \left[\left[(\cos(c) - 1) \cos(9)^4 + (-\cos(c) + 1) \cos(9)^2 + \cos(c), \left((\cos(c) - 1) \cos(9)^2 \right. \right. \right. (22.87) \\ & \left. \left. \left. - \cos(c) + 1 \right) \sin(9) - \sin(c) \right) \cos(9), \sin(9) (-\cos(c) \cos(9)^2 + \sin(c) \sin(9) \right. \right. \\ & \left. \left. + \cos(9)^2) \right], \right. \\ & \left[\left((\cos(c) - 1) \cos(9)^2 - \cos(c) + 1 \right) \sin(9) + \sin(c) \right) \cos(9), 1 + (-\cos(c) \right. \\ & \left. + 1) \cos(9)^4 + (2 \cos(c) - 2) \cos(9)^2, -\sin(9) \cos(9) ((\cos(c) - 1) \sin(9) + \sin(c)) \right. \\ & \left. \right], \\ & \left[-\sin(9) ((\cos(c) - 1) \cos(9)^2 + \sin(c) \sin(9)), \sin(9) \cos(9) (-\cos(c) \sin(9) \right. \\ & \left. + \sin(c) + \sin(9)), (-\cos(c) + 1) \cos(9)^2 + \cos(c) \right] \end{aligned}$$

> map(length, MM);

$$\begin{bmatrix} 85 & 101 & 89 \\ 101 & 77 & 81 \\ 85 & 81 & 49 \end{bmatrix} \quad (22.88)$$

> map(combine, MM);

$$\begin{aligned} & \left[\left[\frac{\cos(c-36)}{16} + \frac{\cos(c+36)}{16} + \frac{7 \cos(c)}{8} - \frac{\cos(36)}{8} + \frac{1}{8}, -\frac{\sin(36)}{8} + \frac{\sin(18)}{4} \right. \right. (22.89) \\ & \left. + \frac{\sin(c+36)}{16} - \frac{\sin(c+18)}{8} + \frac{\sin(c-18)}{8} - \frac{\sin(c-36)}{16} - \frac{\sin(c+9)}{2} \right. \\ & \left. - \frac{\sin(c-9)}{2}, \frac{\sin(c)}{2} - \frac{\sin(c-18)}{4} - \frac{\sin(c+18)}{4} + \frac{\sin(c-9)}{8} \right. \\ & \left. + \frac{\sin(-27+c)}{8} - \frac{\sin(27+c)}{8} - \frac{\sin(c+9)}{8} + \frac{\sin(27)}{4} + \frac{\sin(9)}{4} \right], \\ & \left[-\frac{\sin(36)}{8} + \frac{\sin(18)}{4} + \frac{\sin(c+36)}{16} - \frac{\sin(c+18)}{8} + \frac{\sin(c-18)}{8} \right. \\ & \left. - \frac{\sin(c-36)}{16} + \frac{\sin(c+9)}{2} + \frac{\sin(c-9)}{2}, -\frac{\cos(c-36)}{16} - \frac{\cos(c+36)}{16} \right] \end{aligned}$$

$$\begin{aligned}
& + \frac{5 \cos(c)}{8} + \frac{3}{8} + \frac{\cos(36)}{8} - \frac{\cos(18)}{2} + \frac{\cos(c-18)}{4} + \frac{\cos(c+18)}{4}, \\
& - \frac{\cos(c-9)}{8} + \frac{\cos(27+c)}{8} + \frac{\cos(-27+c)}{8} - \frac{\cos(c+9)}{8} + \frac{\cos(c+18)}{4} \\
& - \frac{\cos(c-18)}{4} + \frac{\cos(9)}{4} - \frac{\cos(27)}{4} \Big], \\
& \Big[- \frac{\sin(c)}{2} + \frac{\sin(c-18)}{4} + \frac{\sin(c+18)}{4} + \frac{\sin(c-9)}{8} + \frac{\sin(-27+c)}{8} \\
& - \frac{\sin(27+c)}{8} - \frac{\sin(c+9)}{8} + \frac{\sin(27)}{4} + \frac{\sin(9)}{4}, - \frac{\cos(c-9)}{8} + \frac{\cos(27+c)}{8} \\
& + \frac{\cos(-27+c)}{8} - \frac{\cos(c+9)}{8} - \frac{\cos(c+18)}{4} + \frac{\cos(c-18)}{4} + \frac{\cos(9)}{4} \\
& - \frac{\cos(27)}{4}, - \frac{\cos(c-18)}{4} - \frac{\cos(c+18)}{4} + \frac{\cos(c)}{2} + \frac{\cos(18)}{2} + \frac{1}{2} \Big] \Big]
\end{aligned}$$

> map(length,%);

$$\begin{bmatrix} 94 & 185 & 198 \\ 185 & 161 & 182 \\ 198 & 182 & 92 \end{bmatrix}$$

(22.90)

> map(factor, MM);

$$\begin{aligned}
& \Big[\Big[\cos(c) \cos(9)^4 - \cos(9)^4 - \cos(c) \cos(9)^2 + \cos(9)^2 + \cos(c), \\
& \cos(9) \left(\sin(9) \cos(c) \cos(9)^2 - \cos(9)^2 \sin(9) - \cos(c) \sin(9) + \sin(9) - \sin(c) \right), \\
& \sin(9) \left(-\cos(c) \cos(9)^2 + \sin(c) \sin(9) + \cos(9)^2 \right) \Big], \\
& \Big[\cos(9) \left(\sin(9) \cos(c) \cos(9)^2 - \cos(9)^2 \sin(9) - \cos(c) \sin(9) + \sin(c) + \sin(9) \right), \\
& -\cos(c) \cos(9)^4 + \cos(9)^4 + 2 \cos(c) \cos(9)^2 - 2 \cos(9)^2 + 1, \\
& -\sin(9) \cos(9) \left(\cos(c) \sin(9) + \sin(c) - \sin(9) \right) \Big], \\
& \Big[-\sin(9) \left(\cos(c) \cos(9)^2 + \sin(c) \sin(9) - \cos(9)^2 \right), -\sin(9) \cos(9) \left(\cos(c) \sin(9) \right. \\
& \left. - \sin(9) - \sin(c) \right), -\cos(c) \cos(9)^2 + \cos(9)^2 + \cos(c) \Big] \Big]
\end{aligned}$$

(22.91)

> subs(cos(a)^2=1-sin(a)^2,%);

$$\begin{aligned}
& \Big[\Big[\cos(c) \cos(9)^4 - \cos(9)^4 - \cos(c) \left(1 - \sin(9)^2 \right) + 1 - \sin(9)^2 + \cos(c), \\
& \cos(9) \left(\sin(9) \cos(c) \left(1 - \sin(9)^2 \right) - \left(1 - \sin(9)^2 \right) \sin(9) - \cos(c) \sin(9) + \sin(9) \right. \\
& \left. - \sin(c) \right), \sin(9) \left(-\cos(c) \left(1 - \sin(9)^2 \right) + \sin(c) \sin(9) + 1 - \sin(9)^2 \right) \Big], \\
& \Big[\cos(9) \left(\sin(9) \cos(c) \left(1 - \sin(9)^2 \right) - \left(1 - \sin(9)^2 \right) \sin(9) - \cos(c) \sin(9) + \sin(c) \right.
\end{aligned}$$

(22.92)

$$\begin{aligned}
& + \sin(9) \big), -\cos(c) \cos(9)^4 + \cos(9)^4 + 2 \cos(c) (1 - \sin(9)^2) - 1 + 2 \sin(9)^2, \\
& -\sin(9) \cos(9) (\cos(c) \sin(9) + \sin(c) - \sin(9)) \big], \\
& \big[-\sin(9) (\cos(c) (1 - \sin(9)^2) + \sin(c) \sin(9) - 1 + \sin(9)^2), \\
& -\sin(9) \cos(9) (\cos(c) \sin(9) - \sin(9) - \sin(c)), -\cos(c) (1 - \sin(9)^2) + 1 \\
& - \sin(9)^2 + \cos(c) \big] \big]
\end{aligned}$$

> map(factor,%);

$$\begin{aligned}
& \big[\big[\cos(c) \cos(9)^4 - \cos(9)^4 + \sin(9)^2 \cos(c) - \sin(9)^2 + 1, -\cos(9) (\sin(9)^3 \cos(c) \\
& - \sin(9)^3 + \sin(c)), \sin(9) (\sin(9)^2 \cos(c) + \sin(c) \sin(9) - \sin(9)^2 - \cos(c) + 1) \\
& \big], \\
& \big[-\cos(9) (\sin(9)^3 \cos(c) - \sin(9)^3 - \sin(c)), -\cos(c) \cos(9)^4 + \cos(9)^4 \\
& - 2 \sin(9)^2 \cos(c) + 2 \sin(9)^2 + 2 \cos(c) - 1, -\sin(9) \cos(9) (\cos(c) \sin(9) + \sin(c) \\
& - \sin(9)) \big], \\
& \big[\sin(9) (\sin(9)^2 \cos(c) - \sin(9)^2 - \sin(c) \sin(9) - \cos(c) + 1), \\
& -\sin(9) \cos(9) (\cos(c) \sin(9) - \sin(9) - \sin(c)), \sin(9)^2 \cos(c) - \sin(9)^2 + 1 \big] \big]
\end{aligned} \tag{22.93}$$

> subs(cos(b)^2=1-sin(b)^2,%);

$$\begin{aligned}
& \big[\big[\cos(c) \cos(9)^4 - \cos(9)^4 + \sin(9)^2 \cos(c) - \sin(9)^2 + 1, -\cos(9) (\sin(9)^3 \cos(c) \\
& - \sin(9)^3 + \sin(c)), \sin(9) (\sin(9)^2 \cos(c) + \sin(c) \sin(9) - \sin(9)^2 - \cos(c) + 1) \\
& \big], \\
& \big[-\cos(9) (\sin(9)^3 \cos(c) - \sin(9)^3 - \sin(c)), -\cos(c) \cos(9)^4 + \cos(9)^4 \\
& - 2 \sin(9)^2 \cos(c) + 2 \sin(9)^2 + 2 \cos(c) - 1, -\sin(9) \cos(9) (\cos(c) \sin(9) + \sin(c) \\
& - \sin(9)) \big], \\
& \big[\sin(9) (\sin(9)^2 \cos(c) - \sin(9)^2 - \sin(c) \sin(9) - \cos(c) + 1), \\
& -\sin(9) \cos(9) (\cos(c) \sin(9) - \sin(9) - \sin(c)), \sin(9)^2 \cos(c) - \sin(9)^2 + 1 \big] \big]
\end{aligned} \tag{22.94}$$

> map(factor,%);

$$\begin{aligned}
& \big[\big[\cos(c) \cos(9)^4 - \cos(9)^4 + \sin(9)^2 \cos(c) - \sin(9)^2 + 1, -\cos(9) (\sin(9)^3 \cos(c) \\
& - \sin(9)^3 + \sin(c)), \sin(9) (\sin(9)^2 \cos(c) + \sin(c) \sin(9) - \sin(9)^2 - \cos(c) + 1) \\
& \big], \\
& \big[-\cos(9) (\sin(9)^3 \cos(c) - \sin(9)^3 - \sin(c)), -\cos(c) \cos(9)^4 + \cos(9)^4 \\
& - 2 \sin(9)^2 \cos(c) + 2 \sin(9)^2 + 2 \cos(c) - 1, -\sin(9) \cos(9) (\cos(c) \sin(9) + \sin(c)
\end{aligned} \tag{22.95}$$

$$\begin{aligned} & -\sin(9))], \\ & [\sin(9) (\sin(9)^2 \cos(c) - \sin(9)^2 - \sin(c) \sin(9) - \cos(c) + 1), \\ & -\sin(9) \cos(9) (\cos(c) \sin(9) - \sin(9) - \sin(c)), \sin(9)^2 \cos(c) - \sin(9)^2 + 1]] \end{aligned}$$

> map(length,%);

$$\begin{bmatrix} 93 & 77 & 105 \\ 77 & 105 & 85 \\ 105 & 85 & 49 \end{bmatrix} \quad (22.96)$$

> gp_to_q(p1);

$$\begin{aligned} & Q\left(\cos\left(\frac{\pi}{12}\right), \frac{\sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sqrt{2} \sin\left(\frac{\pi}{12}\right)}{2}, \right. \\ & \left. - \frac{\sin\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sqrt{2} \sin\left(\frac{\pi}{12}\right)}{2}, -\cos\left(2 \arccos\left(\frac{\sqrt{3}}{3}\right)\right) \sin\left(\frac{\pi}{12}\right) \right) \end{aligned} \quad (22.97)$$

> map(expand,%);

$$Q\left(\cos\left(\frac{\pi}{12}\right), \frac{\sqrt{2} \sin\left(\frac{\pi}{12}\right) \sqrt{6} \sqrt{3}}{9}, -\frac{\sqrt{2} \sin\left(\frac{\pi}{12}\right) \sqrt{6} \sqrt{3}}{9}, \frac{\sin\left(\frac{\pi}{12}\right)}{3}\right) \quad (22.98)$$

> map(simplify,%);

$$Q\left(\cos\left(\frac{\pi}{12}\right), \frac{2 \sin\left(\frac{\pi}{12}\right)}{3}, -\frac{2 \sin\left(\frac{\pi}{12}\right)}{3}, \frac{\sin\left(\frac{\pi}{12}\right)}{3}\right) \quad (22.99)$$

> p2;

$$p2 \quad (22.100)$$

> p2 := PHYS_ROT(a,b,c);

$$p2 := \text{PHYS_ROT}(9, 9, c) \quad (22.101)$$

> gp_to_q(p2);

$$Q\left(\cos\left(\frac{c}{2}\right), \cos(9) \sin(9) \sin\left(\frac{c}{2}\right), \sin\left(\frac{c}{2}\right) \sin(9)^2, \cos(9) \sin\left(\frac{c}{2}\right)\right) \quad (22.102)$$

> q2 := %;

$$q2 := Q\left(\cos\left(\frac{c}{2}\right), \cos(9) \sin(9) \sin\left(\frac{c}{2}\right), \sin\left(\frac{c}{2}\right) \sin(9)^2, \cos(9) \sin\left(\frac{c}{2}\right)\right) \quad (22.103)$$

> gp2 := q_to_gp(q2);

$$(22.104)$$

$$gp2 := PHYS_ROT \left(\arccos \left(\frac{\cos(9) \sin\left(\frac{c}{2}\right)}{\sqrt{1 - \cos\left(\frac{c}{2}\right)^2}} \right), \arctan \left(\sin\left(\frac{c}{2}\right) \sin(9)^2, \right. \right. \\ \left. \left. \cos(9) \sin(9) \sin\left(\frac{c}{2}\right) \right), 2 \arccos \left(\cos\left(\frac{c}{2}\right) \right) \right) \quad (22.104)$$

> Mx := <<1 | 0 | 0>, <0 | 0 | 1>, <0 | -1 | 0>>;

$$M_x := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \quad (22.105)$$

> is_mat(Mx);

true (22.106)

> My := <<0 | 0 | -1>, <0 | 1 | 0>, <1 | 0 | 0>>;

$$M_y := \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.107)$$

> is_mat(My);

true (22.108)

> Mz := <<0 | 1 | 0>, <-1 | 0 | 0>, <0 | 0 | 1>>;

$$M_z := \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (22.109)$$

> is_mat(Mz);

true (22.110)

> Mxy := Mx . My;

$$M_{xy} := \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad (22.111)$$

> Myx := My . Mx;

(22.112)

$$M_{yx} := \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.112)$$

> Mxz := Mx . Mz;

$$M_{xz} := \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.113)$$

> Myz := My . Mz;

$$M_{yz} := \begin{bmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (22.114)$$

> Mzy := Mz . My;

$$M_{zy} := \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.115)$$

> Mzx := Mz . Mx;

$$M_{zx} := \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad (22.116)$$

> Mxx := Mx . Mx;

$$M_{xx} := \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (22.117)$$

> Myy := My . My;

$$M_{yy} := \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (22.118)$$

> Mzz := Mz . Mz;

$$M_{zz} := \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (22.119)$$

> Qx := mat_to_q(Mx);

$$Q_x := Q\left(\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}, 0, 0\right) \quad (22.120)$$

> Qy := mat_to_q(My);

$$Q_y := Q\left(\frac{\sqrt{2}}{2}, 0, -\frac{\sqrt{2}}{2}, 0\right) \quad (22.121)$$

> Qz := mat_to_q(Mz);

$$Q_z := Q\left(\frac{\sqrt{2}}{2}, 0, 0, -\frac{\sqrt{2}}{2}\right) \quad (22.122)$$

> Qzy := Qz &** Qy;

$$Q_{zy} := Q\left(\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}\right) \quad (22.123)$$

> q_to_mat(Qzy);

$$\left[\left[\frac{1}{2} + \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{3\sqrt{4}}{8} + \frac{1}{4} \right. \right. \\ \left. \left. - \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, \frac{1}{4} - \frac{3\sqrt{4}}{8} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} \right], \right. \\ \left[\frac{1}{4} - \frac{3\sqrt{4}}{8} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{1}{2} + \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} \right. \\ \left. \left. - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{3\sqrt{4}}{8} + \frac{1}{4} - \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} \right], \right. \\ \left[\frac{3\sqrt{4}}{8} + \frac{1}{4} - \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, \frac{1}{4} - \frac{3\sqrt{4}}{8} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{1}{2} \right. \\ \left. \left. + \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} \right] \right] \quad (22.124)$$

> map(simplify, %);

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.125)$$

> Qxy := Qx &** Qy;

$$Q_{xy} := Q\left(\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}, \frac{1}{2}\right) \quad (22.126)$$

> q_to_mat(Qxy);

$$\begin{aligned} & \left[\left[\frac{1}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2}, \frac{1}{4} - \frac{3\sqrt{4}}{8} \right. \right. \\ & \quad \left. \left. - \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, -\frac{1}{4} - \frac{3\sqrt{4}}{8} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2} \right], \right. \\ & \quad \left[\frac{3\sqrt{4}}{8} + \frac{1}{4} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2}, \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} + \frac{1}{2} \right. \\ & \quad \left. + \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} + \frac{3\sqrt{4}}{8} - \frac{1}{4} \right], \\ & \quad \left[\frac{3\sqrt{4}}{8} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} - \frac{1}{4}, \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{3\sqrt{4}}{8} - \frac{1}{4}, \right. \\ & \quad \left. \left. \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2} + \frac{1}{2} \right] \right] \end{aligned} \quad (22.127)$$

> map(simplify, %) = Mxy;

$$\begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad (22.128)$$

> Qxz := Qx & Qz;**

$$Q_{xz} := Q\left(\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}\right) \quad (22.129)$$

> Rxz := q_to_mat(Qxz);

$$\begin{aligned} R_{xz} := & \left[\left[\frac{1}{2} + \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{3\sqrt{4}}{8} + \frac{1}{4} \right. \right. \\ & \left. \left. - \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, \frac{1}{4} - \frac{3\sqrt{4}}{8} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} \right], \right. \end{aligned} \quad (22.130)$$

$$\left[\frac{1}{4} - \frac{3\sqrt{4}}{8} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{1}{2} + \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} \right. \\ \left. - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{3\sqrt{4}}{8} + \frac{1}{4} - \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} \right], \\ \left[\frac{3\sqrt{4}}{8} + \frac{1}{4} - \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, \frac{1}{4} - \frac{3\sqrt{4}}{8} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{1}{2} \right. \\ \left. + \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} \right] \right]$$

> map(simplify, Rxz) = Mxz;

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.131)$$

> Qzx := Qz & Qx;**

$$Q_{zx} := Q\left(\frac{1}{2}, -\frac{1}{2}, \frac{1}{2}, -\frac{1}{2}\right) \quad (22.132)$$

> Rxz := q_to_mat(Qzx);

$$R_{zx} := \left[\left[\frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} + \frac{1}{2} + \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} \right. \right. \\ \left. \left. + \frac{3\sqrt{4}}{8} - \frac{1}{4}, \frac{3\sqrt{4}}{8} + \frac{1}{4} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2} \right], \right. \\ \left[\frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{3\sqrt{4}}{8} - \frac{1}{4}, \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2} \right. \\ \left. \left. + \frac{1}{2}, \frac{3\sqrt{4}}{8} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} - \frac{1}{4} \right], \right. \\ \left[\frac{1}{4} - \frac{3\sqrt{4}}{8} - \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, -\frac{1}{4} - \frac{3\sqrt{4}}{8} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2}, \frac{1}{2} \right] \right]$$

(22.133)

$$\left. - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2} \right] \right]$$

> map(simplify, Rzx) = Mzx;

$$\begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad (22.134)$$

> Qyx := Qy & Qx;**

$$Q_{yx} := Q\left(\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}\right) \quad (22.135)$$

> Ryx := q_to_mat(Qyx);

$$\begin{aligned} Ryx := & \left[\left[\frac{1}{2} + \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{3\sqrt{4}}{8} + \frac{1}{4} \right. \right. \\ & \left. \left. - \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, \frac{1}{4} - \frac{3\sqrt{4}}{8} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} \right], \right. \\ & \left[\frac{1}{4} - \frac{3\sqrt{4}}{8} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{1}{2} + \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} \right. \\ & \left. \left. - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{3\sqrt{4}}{8} + \frac{1}{4} - \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} \right], \right. \\ & \left[\frac{3\sqrt{4}}{8} + \frac{1}{4} - \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2}, \frac{1}{4} - \frac{3\sqrt{4}}{8} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2}, \frac{1}{2} \right. \\ & \left. \left. + \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} \right] \right] \end{aligned} \quad (22.136)$$

> map(simplify, Ryx) = Myx;

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.137)$$

> Qyz := Qy & Qz;**

(22.138)

$$Q_{yz} := Q\left(\frac{1}{2}, \frac{1}{2}, -\frac{1}{2}, -\frac{1}{2}\right) \quad (22.138)$$

> Ryz := q_to_mat(Qyz);

$$R_{yz} := \left[\left[\frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2} + \frac{1}{2}, \frac{3\sqrt{4}}{8} \right. \right. \\ \left. \left. - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} - \frac{1}{4}, \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} - \frac{3\sqrt{4}}{8} - \frac{1}{4} \right], \right. \\ \left[-\frac{1}{4} - \frac{3\sqrt{4}}{8} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2}, \frac{1}{2} - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} + \frac{1}{4} \right)}{2} \right. \\ \left. \left. - \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2}, \frac{1}{4} - \frac{3\sqrt{4}}{8} - \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} \right], \right. \\ \left[\frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} + \frac{3\sqrt{4}}{8} - \frac{1}{4}, \frac{3\sqrt{4}}{8} + \frac{1}{4} + \frac{\sqrt{4} \left(\frac{\sqrt{4}}{8} - \frac{1}{4} \right)}{2}, \right. \\ \left. \left. \frac{\sqrt{4} \left(\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} + \frac{1}{2} + \frac{\sqrt{4} \left(-\frac{1}{4} - \frac{\sqrt{4}}{8} \right)}{2} \right] \right] \quad (22.139)$$

> map(simplify, Ryz) = Myz;

$$\begin{bmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (22.140)$$

> Px := mat_to_gp(Mx);

$$Px := PHYS_ROT\left(\frac{\pi}{2}, \pi, \frac{\pi}{2}\right) \quad (22.141)$$

> Py := mat_to_gp(My);

$$Py := PHYS_ROT\left(\frac{\pi}{2}, -\frac{\pi}{2}, \frac{\pi}{2}\right) \quad (22.142)$$

> Pz := mat_to_gp(Mz);

$$Pz := PHYS_ROT\left(\pi, 0, \frac{\pi}{2}\right) \quad (22.143)$$

> Pxy := gp_mul(Px,Py);

$$P_{xy} := PHYS_ROT\left(\arccos\left(\frac{\sqrt{3}\sqrt{4}}{6}\right), -\frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.144)$$

> Pxy := map(simplify, Pxy);

$$P_{xy} := PHYS_ROT\left(\arccos\left(\frac{\sqrt{3}}{3}\right), -\frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.145)$$

> gp_to_mat(Pxy);

$$\begin{bmatrix} 0 & 0 & -\frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} \\ 1 & 0 & 0 \\ 0 & -\frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} & 0 \end{bmatrix} \quad (22.146)$$

> map(simplify, %) = Mxy;

$$\begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad (22.147)$$

> Pyx := gp_mul(Py, Px);

$$P_{yx} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}\sqrt{4}}{6}\right), -\frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.148)$$

> Pyx := map(simplify, Pyx);

$$P_{yx} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}}{3}\right), -\frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.149)$$

> gp_to_mat(Pyx);

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} \\ \frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} & 0 & 0 \end{bmatrix} \quad (22.150)$$

> map(simplify, %) = Myx;

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.151)$$

> Pxz := gp_mul(Px, Pz);

(22.152)

$$P_{xz} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}\sqrt{4}}{6}\right), -\frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.152)$$

> Pxz := map(simplify, Pxz);

$$P_{xz} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}}{3}\right), -\frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.153)$$

> gp_to_mat(Pxz);

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} \\ \frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} & 0 & 0 \end{bmatrix} \quad (22.154)$$

> map(simplify, %) = Mxz;

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.155)$$

> Pzx := gp_mul(Pz,Px);

$$P_{zx} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}\sqrt{4}}{6}\right), \frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.156)$$

> Pzx := map(simplify, Pzx);

$$P_{zx} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}}{3}\right), \frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.157)$$

> gp_to_mat(Pzx);

$$\begin{bmatrix} 0 & 0 & \frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} \\ -1 & 0 & 0 \\ 0 & -\frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} & 0 \end{bmatrix} \quad (22.158)$$

> map(simplify, %) = Mzx;

$$\begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad (22.159)$$

> Pyz := gp_mul(Py,Pz);

(22.160)

$$P_{yz} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}\sqrt{4}}{6}\right), -\frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.160)$$

> Pyz := map(simplify, Pyz);

$$P_{yz} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}}{3}\right), -\frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.161)$$

> gp_to_mat(Pyz);

$$\begin{bmatrix} 0 & 0 & -\frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} \\ -1 & 0 & 0 \\ 0 & \frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} & 0 \end{bmatrix} \quad (22.162)$$

> map(simplify, %) = Myz;

$$\begin{bmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (22.163)$$

> Pzy := gp_mul(Pz,Py);

$$P_{zy} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}\sqrt{4}}{6}\right), -\frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.164)$$

> Pzy := map(simplify, Pzy);

$$P_{zy} := PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}}{3}\right), -\frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.165)$$

> gp_to_mat(Pzy);

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} \\ \frac{\sqrt{2}\sqrt{6}\sqrt{3}}{6} & 0 & 0 \end{bmatrix} \quad (22.166)$$

> map(simplify, %) = Mzy;

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (22.167)$$

> Pxx := gp_mul(Px,Px);

(22.168)

$$P_{xx} := PHYS_ROT\left(\frac{\pi}{2}, \pi, \pi\right) \quad (22.168)$$

> Pxx := map(simplify, Pxx);

$$P_{xx} := PHYS_ROT\left(\frac{\pi}{2}, \pi, \pi\right) \quad (22.169)$$

> gp_to_mat(Pxx);

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (22.170)$$

> map(simplify, %) = Mxx;

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (22.171)$$

> Pyy := gp_mul(Py,Py);

$$P_{yy} := PHYS_ROT\left(\frac{\pi}{2}, -\frac{\pi}{2}, \pi\right) \quad (22.172)$$

> Pyy := map(simplify, Pyy);

$$P_{yy} := PHYS_ROT\left(\frac{\pi}{2}, -\frac{\pi}{2}, \pi\right) \quad (22.173)$$

> gp_to_mat(Pyy);

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (22.174)$$

> map(simplify, %) = Myy;

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (22.175)$$

> Pzz := gp_mul(Pz,Pz);

$$P_{zz} := PHYS_ROT(\pi, 0, \pi) \quad (22.176)$$

> Pzz := map(simplify, Pzz);

$$P_{zz} := PHYS_ROT(\pi, 0, \pi) \quad (22.177)$$

> gp_to_mat(Pzz);

$$(22.178)$$

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (22.178)$$

> map(simplify, %) = Myy;

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (22.179)$$

> mat_to_q(Myx);

$$Q(0, 0, 1, 0) \quad (22.180)$$

> q_to_mat(%);

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (22.181)$$

> mat_to_gp(Myx);

$$PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}}{3}\right), -\frac{\pi}{4}, \frac{2\pi}{3}\right) \quad (22.182)$$

> map(simplify, gp_to_mat(%)) = Myz;

$$\begin{bmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (22.183)$$

> mat_to_gp(Mzx);

$$PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}}{3}\right), \frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.184)$$

> map(simplify, gp_to_q(%));

$$Q\left(\frac{1}{2}, -\frac{1}{2}, \frac{1}{2}, -\frac{1}{2}\right) \quad (22.185)$$

> map(simplify, q_to_gp(%));

$$PHYS_ROT\left(\pi - \arccos\left(\frac{\sqrt{3}}{3}\right), \frac{3\pi}{4}, \frac{2\pi}{3}\right) \quad (22.186)$$

> Q(-7,-9,11,-13);

$$Q(-7, -9, 11, -13) \quad (22.187)$$

> Qnorm(%);

$$\sqrt{420} \quad (22.188)$$

> %%& // Q(%, 0, 0, 0);

$$Q\left(-\frac{\sqrt{420}}{60}, -\frac{3\sqrt{420}}{140}, \frac{11\sqrt{420}}{420}, -\frac{13\sqrt{420}}{420}\right) \quad (22.189)$$

> map(simplify, %);

$$Q\left(-\frac{\sqrt{105}}{30}, -\frac{3\sqrt{105}}{70}, \frac{11\sqrt{105}}{210}, -\frac{13\sqrt{105}}{210}\right) \quad (22.190)$$

> map(simplify, q_to_gp(%));

$$PHYS_ROT\left(\pi - \arccos\left(\frac{13\sqrt{371}}{371}\right), -\arctan\left(\frac{11}{9}\right) + \pi, 2\pi - 2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \quad (22.191)$$

> gp_to_q(%);

$$Q\left(-\frac{\sqrt{105}}{30}, -\frac{3\sqrt{74942}\sqrt{202}\sqrt{795}}{749420}, \frac{11\sqrt{74942}\sqrt{202}\sqrt{795}}{2248260}, -\frac{13\sqrt{371}\sqrt{795}}{11130}\right) \quad (22.192)$$

> map(combine, %);

$$Q\left(-\frac{\sqrt{105}}{30}, -\frac{3\sqrt{12034935780}}{749420}, \frac{11\sqrt{12034935780}}{2248260}, -\frac{13\sqrt{294945}}{11130}\right) \quad (22.193)$$

> q_to_mat(%);

$$\left[\left[\frac{13}{42} + \frac{13\sqrt{60}\sqrt{7}\sqrt{294945}\left(-\frac{13\sqrt{60}\sqrt{7}\sqrt{294945}}{667800} - \frac{33}{140}\right)}{77910} \right. \right. \quad (22.194)$$

$$+ \frac{1}{15737820} \left(11\sqrt{60}\sqrt{7}\sqrt{12034935780} \left(\frac{13\sqrt{294945}\sqrt{12034935780}}{2780348200} \right. \right. \\ \left. \left. - \frac{11\sqrt{60}\sqrt{7}\sqrt{12034935780}}{134895600} \right) \right), \frac{949\sqrt{60}\sqrt{7}\sqrt{294945}}{10907400} - \frac{33}{140}$$

$$+ \frac{1}{15737820} \left(11\sqrt{60}\sqrt{7}\sqrt{12034935780} \left(-\frac{\sqrt{60}\sqrt{7}\sqrt{12034935780}}{14988400} \right. \right. \\ \left. \left. - \frac{143\sqrt{294945}\sqrt{12034935780}}{25023133800} \right) \right), \frac{13\sqrt{294945}\sqrt{12034935780}}{2780348200}$$

$$+ \frac{979\sqrt{60}\sqrt{7}\sqrt{12034935780}}{2203294800}$$

$$\begin{aligned}
& + \frac{1}{77910} \left(13 \sqrt{60} \sqrt{7} \sqrt{294945} \left(\frac{\sqrt{60} \sqrt{7} \sqrt{12034935780}}{14988400} \right. \right. \\
& \left. \left. - \frac{143 \sqrt{294945} \sqrt{12034935780}}{25023133800} \right) \right) \Bigg], \\
& \left[- \frac{2327 \sqrt{60} \sqrt{7} \sqrt{294945}}{32722200} - \frac{33}{140} \right. \\
& + \frac{1}{5245940} \left(3 \sqrt{60} \sqrt{7} \sqrt{12034935780} \left(\frac{13 \sqrt{294945} \sqrt{12034935780}}{2780348200} \right. \right. \\
& \left. \left. - \frac{11 \sqrt{60} \sqrt{7} \sqrt{12034935780}}{134895600} \right) \right) \Bigg], \\
& - \frac{13 \sqrt{60} \sqrt{7} \sqrt{294945} \left(\frac{13 \sqrt{60} \sqrt{7} \sqrt{294945}}{667800} - \frac{33}{140} \right)}{77910} + \frac{17}{42} \\
& + \frac{1}{5245940} \left(3 \sqrt{60} \sqrt{7} \sqrt{12034935780} \left(- \frac{\sqrt{60} \sqrt{7} \sqrt{12034935780}}{14988400} \right. \right. \\
& \left. \left. - \frac{143 \sqrt{294945} \sqrt{12034935780}}{25023133800} \right) \right) \Bigg], \\
& - \frac{1}{77910} \left(13 \sqrt{60} \sqrt{7} \sqrt{294945} \left(\frac{13 \sqrt{294945} \sqrt{12034935780}}{2780348200} \right. \right. \\
& + \frac{11 \sqrt{60} \sqrt{7} \sqrt{12034935780}}{134895600} \left. \right) \Bigg) + \frac{267 \sqrt{60} \sqrt{7} \sqrt{12034935780}}{734431600} \\
& \left. - \frac{143 \sqrt{294945} \sqrt{12034935780}}{25023133800} \right], \\
& \left[- \frac{1969 \sqrt{60} \sqrt{7} \sqrt{12034935780}}{6609884400} \right. \\
& \left. - \frac{3 \sqrt{60} \sqrt{7} \sqrt{12034935780} \left(- \frac{13 \sqrt{60} \sqrt{7} \sqrt{294945}}{667800} - \frac{33}{140} \right)}{5245940} \right]
\end{aligned}$$

$$\begin{aligned}
& + \frac{13 \sqrt{294945} \sqrt{12034935780}}{2780348200}, \\
& - \frac{11 \sqrt{60} \sqrt{7} \sqrt{12034935780} \left(\frac{13 \sqrt{60} \sqrt{7} \sqrt{294945}}{667800} - \frac{33}{140} \right)}{15737820} \\
& - \frac{219 \sqrt{60} \sqrt{7} \sqrt{12034935780}}{734431600} - \frac{143 \sqrt{294945} \sqrt{12034935780}}{25023133800}, \\
& - \frac{1}{15737820} \left(11 \sqrt{60} \sqrt{7} \sqrt{12034935780} \left(\frac{13 \sqrt{294945} \sqrt{12034935780}}{2780348200} \right. \right. \\
& \left. \left. + \frac{11 \sqrt{60} \sqrt{7} \sqrt{12034935780}}{134895600} \right) \right) \\
& - \frac{1}{5245940} \left(3 \sqrt{60} \sqrt{7} \sqrt{12034935780} \left(\frac{\sqrt{60} \sqrt{7} \sqrt{12034935780}}{14988400} \right. \right. \\
& \left. \left. - \frac{143 \sqrt{294945} \sqrt{12034935780}}{25023133800} \right) \right) + \frac{109}{210} \Bigg]
\end{aligned}$$

> map(simplify, %);

$$\begin{bmatrix} -\frac{8}{21} & -\frac{4}{105} & \frac{97}{105} \\ -\frac{19}{21} & -\frac{4}{21} & -\frac{8}{21} \\ \frac{4}{21} & -\frac{103}{105} & \frac{4}{105} \end{bmatrix} \quad (22.195)$$

> mat_to_q(%);

$$Q\left(\frac{\sqrt{15} \sqrt{7}}{30}, -\frac{3 \sqrt{60} \sqrt{7}}{140}, \frac{11 \sqrt{60} \sqrt{7}}{420}, -\frac{13 \sqrt{60} \sqrt{7}}{420}\right) \quad (22.196)$$

> map(combine, %);

$$Q\left(\frac{\sqrt{105}}{30}, -\frac{3 \sqrt{420}}{140}, \frac{11 \sqrt{420}}{420}, -\frac{13 \sqrt{420}}{420}\right) \quad (22.197)$$

> q_to_gp(%);

$$PHYS_ROT\left(\pi - \arccos\left(\frac{13 \sqrt{420} \sqrt{53} \sqrt{60}}{22260}\right), -\arctan\left(\frac{11}{9}\right) + \pi, 2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \quad (22.198)$$

> map(simplify, %);

$$PHYS_ROT\left(\pi - \arccos\left(\frac{13\sqrt{371}}{371}\right), -\arctan\left(\frac{11}{9}\right) + \pi, 2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \quad (22.199)$$

> map(combine, %);

$$PHYS_ROT\left(\pi - \arccos\left(\frac{13\sqrt{371}}{371}\right), -\arctan\left(\frac{11}{9}\right) + \pi, 2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \quad (22.200)$$

> gp_to_mat(%);

$$\begin{aligned} & \left[\left[\begin{aligned} & -\frac{1}{202} \left(9\sqrt{202} \left(-\frac{1}{371} \left(13\sqrt{371} \left(\frac{117 \cos\left(2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{371} \sqrt{202}}{74942} \right. \right. \right. \right. \right. \\ & \left. \left. \left. + \frac{11 \sin\left(2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{202}}{202} \right) \right) - \frac{9\sqrt{202}}{371} \right) \right) \right. \\ & \left. - \frac{1}{202} \left(11\sqrt{202} \left(\frac{117 \sin\left(2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{371} \sqrt{202}}{74942} \right. \right. \right. \right. \right. \\ & \left. \left. \left. - \frac{11 \cos\left(2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{202}}{202} \right) \right) \right), -\frac{1}{202} \left(9\sqrt{202} \left(\right. \right. \right. \\ & \left. \left. \left. - \frac{1}{371} \left(13\sqrt{371} \left(-\frac{143 \cos\left(2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{371} \sqrt{202}}{74942} \right. \right. \right. \right. \right. \\ & \left. \left. \left. + \frac{9 \sin\left(2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{202}}{202} \right) \right) \right) + \frac{11\sqrt{202}}{371} \right) \right) - \frac{1}{202} \left(11\sqrt{202} \left(\right. \right. \right. \\ & \left. \left. \left. - \frac{143 \sin\left(2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{371} \sqrt{202}}{74942} - \frac{9 \cos\left(2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{202}}{202} \right) \right) \right) \end{aligned} \right] \end{aligned} \quad (22.201)$$

$$\begin{aligned}
& , - \frac{9 \sqrt{202} \left(\frac{13 \sqrt{371} \cos \left(2 \arccos \left(\frac{\sqrt{105}}{30} \right) \right) \sqrt{74942}}{137641} - \frac{13 \sqrt{74942} \sqrt{371}}{137641} \right)}{202} \\
& + \frac{11 \sqrt{202} \sin \left(2 \arccos \left(\frac{\sqrt{105}}{30} \right) \right) \sqrt{74942}}{74942} \Bigg], \\
& \left[\frac{1}{202} \left(11 \sqrt{202} \left(\right. \right. \right. \\
& - \frac{1}{371} \left(13 \sqrt{371} \left(\frac{117 \cos \left(2 \arccos \left(\frac{\sqrt{105}}{30} \right) \right) \sqrt{371} \sqrt{202}}{74942} \right. \right. \right. \\
& + \left. \left. \left. \frac{11 \sin \left(2 \arccos \left(\frac{\sqrt{105}}{30} \right) \right) \sqrt{202}}{202} \right) \right) - \frac{9 \sqrt{202}}{371} \right) \right) \\
& - \frac{1}{202} \left(9 \sqrt{202} \left(\frac{117 \sin \left(2 \arccos \left(\frac{\sqrt{105}}{30} \right) \right) \sqrt{371} \sqrt{202}}{74942} \right. \right. \right. \\
& - \left. \left. \left. \frac{11 \cos \left(2 \arccos \left(\frac{\sqrt{105}}{30} \right) \right) \sqrt{202}}{202} \right) \right) , \frac{1}{202} \left(11 \sqrt{202} \left(\right. \right. \right. \\
& - \frac{1}{371} \left(13 \sqrt{371} \left(- \frac{143 \cos \left(2 \arccos \left(\frac{\sqrt{105}}{30} \right) \right) \sqrt{371} \sqrt{202}}{74942} \right. \right. \right.
\end{aligned}$$

$$\begin{aligned}
& + \frac{9 \sin\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{202}}{202} \Bigg) + \frac{11 \sqrt{202}}{371} \Bigg) - \frac{1}{202} \left(9 \sqrt{202} \left(\right. \right. \\
& - \frac{143 \sin\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{371} \sqrt{202}}{74942} - \frac{9 \cos\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{202}}{202} \Bigg) \\
& \left. \left. \frac{11 \sqrt{202} \left(\frac{13 \sqrt{371} \cos\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{74942}}{137641} - \frac{13 \sqrt{74942} \sqrt{371}}{137641} \right)}{202} \right. \right. \\
& \left. \left. + \frac{9 \sqrt{202} \sin\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{74942}}{74942} \right] \right. \\
& \left[- \frac{1}{371} \left(\sqrt{74942} \left(- \frac{117 \cos\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{371} \sqrt{202}}{74942} \right. \right. \right. \\
& \left. \left. + \frac{11 \sin\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{202}}{202} \right) \right) + \frac{117 \sqrt{371} \sqrt{74942} \sqrt{202}}{27803482}, \right. \\
& - \frac{1}{371} \left(\sqrt{74942} \left(- \frac{143 \cos\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{371} \sqrt{202}}{74942} \right. \right. \\
& \left. \left. + \frac{9 \sin\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right) \sqrt{202}}{202} \right) \right) - \frac{143 \sqrt{371} \sqrt{74942} \sqrt{202}}{27803482}, \\
& \left. \left. \frac{202 \cos\left(2 \arccos\left(\frac{\sqrt{105}}{30}\right)\right)}{371} + \frac{169}{371} \right] \right]
\end{aligned}$$

```
> map(simplify, %);
```

$$\begin{bmatrix} -\frac{8}{21} & -\frac{4}{105} & \frac{97}{105} \\ -\frac{19}{21} & -\frac{4}{21} & -\frac{8}{21} \\ \frac{4}{21} & -\frac{103}{105} & \frac{4}{105} \end{bmatrix} \quad (22.202)$$

> map(expand, %);

$$\begin{bmatrix} -\frac{8}{21} & -\frac{4}{105} & \frac{97}{105} \\ -\frac{19}{21} & -\frac{4}{21} & -\frac{8}{21} \\ \frac{4}{21} & -\frac{103}{105} & \frac{4}{105} \end{bmatrix} \quad (22.203)$$

> map(combine, %);

$$\begin{bmatrix} -\frac{8}{21} & -\frac{4}{105} & \frac{97}{105} \\ -\frac{19}{21} & -\frac{4}{21} & -\frac{8}{21} \\ \frac{4}{21} & -\frac{103}{105} & \frac{4}{105} \end{bmatrix} \quad (22.204)$$

> mat_to_gp(%);

$$PHYS_ROT\left(\pi - \arccos\left(\frac{13\sqrt{7}\sqrt{53}}{371}\right), -\arctan\left(\frac{11}{9}\right) + \pi, 2\arccos\left(\frac{\sqrt{15}\sqrt{7}}{30}\right)\right) \quad (22.205)$$

> map(combine, %);

$$PHYS_ROT\left(\pi - \arccos\left(\frac{13\sqrt{371}}{371}\right), -\arctan\left(\frac{11}{9}\right) + \pi, 2\arccos\left(\frac{\sqrt{105}}{30}\right)\right) \quad (22.206)$$

> m313 := ea_to_mat(EA(psi,theta,phi),[3,1,3]);

$$\begin{aligned} m313 := & \left[\left[\cos(\psi) \cos(\phi) - \sin(\psi) \cos(\theta) \sin(\phi), \cos(\psi) \sin(\phi) \right. \right. \\ & \left. \left. + \sin(\psi) \cos(\theta) \cos(\phi), \sin(\psi) \sin(\theta) \right], \right. \\ & \left[-\sin(\psi) \cos(\phi) - \cos(\psi) \cos(\theta) \sin(\phi), -\sin(\psi) \sin(\phi) \right. \\ & \left. + \cos(\psi) \cos(\theta) \cos(\phi), \cos(\psi) \sin(\theta) \right], \\ & \left. \left[\sin(\theta) \sin(\phi), -\sin(\theta) \cos(\phi), \cos(\theta) \right] \right] \end{aligned} \quad (22.207)$$

> ea_to_mat(EA(phi,theta,psi),[1,2,3]);

$$\left[\left[\cos(\theta) \cos(\phi), \cos(\theta) \sin(\phi), -\sin(\theta) \right], \right. \quad (22.208)$$

$$\begin{aligned} & [\sin(\psi) \sin(\theta) \cos(\phi) - \cos(\psi) \sin(\phi), \sin(\psi) \sin(\theta) \sin(\phi) + \cos(\psi) \cos(\phi), \\ & \sin(\psi) \cos(\theta)], \\ & [\cos(\psi) \sin(\theta) \cos(\phi) + \sin(\psi) \sin(\phi), \cos(\psi) \sin(\theta) \sin(\phi) - \sin(\psi) \cos(\phi), \\ & \cos(\psi) \cos(\theta)] \end{aligned}$$

```
> ea_to_mat(EA(psi,theta,phi),[3,2,3]);
```

$$\begin{aligned} & [[-\sin(\psi) \sin(\phi) + \cos(\psi) \cos(\theta) \cos(\phi), \cos(\psi) \cos(\theta) \sin(\phi) + \sin(\psi) \cos(\phi), \\ & -\cos(\psi) \sin(\theta)], \end{aligned} \quad (22.209)$$

$$\begin{aligned} & [-\sin(\psi) \cos(\theta) \cos(\phi) - \cos(\psi) \sin(\phi), \cos(\psi) \cos(\phi) - \sin(\psi) \cos(\theta) \sin(\phi), \\ & \sin(\psi) \sin(\theta)], \\ & [\sin(\theta) \cos(\phi), \sin(\theta) \sin(\phi), \cos(\theta)] \end{aligned}$$

```
> map(simplify, mat_to_ea(m313,[3,1,3]), symbolic);
```

$$EA(\arctan(\sin(\psi), \cos(\psi)), \theta, \arctan(\sin(\phi), \cos(\phi))) \quad (22.210)$$

```
> ea := EA(-0.1,0.3,0.5);
```

$$ea := EA(-0.1, 0.3, 0.5) \quad (22.211)$$

```
> ma := ea_to_mat(ea,[3,1,3]);
```

$$ma := \begin{bmatrix} 0.918923278276532 & 0.393331404755786 & -0.0295027919239752 \\ -0.368112489490253 & 0.882060892042616 & 0.294043836596817 \\ 0.141679934264331 & -0.259343380089004 & 0.955336489100000 \end{bmatrix} \quad (22.212)$$

```
> ea2 := mat_to_ea(ma,[3,1,3]);
```

$$ea2 := EA(-0.100000000000962, 0.3000000000086647, 0.499999999991696) \quad (22.213)$$

```
> ea_to_mat(ea2,[3,1,3]);
```

$$\begin{bmatrix} 0.918923278250237 & 0.393331404744531 & -0.0295027919277252 \\ -0.368112489479719 & 0.882060892017376 & 0.294043836634191 \\ 0.141679934284570 & -0.259343380126051 & 0.955336489100000 \end{bmatrix} \quad (22.214)$$

3.94 quaternionic quantum mechanics of harmonic oscillator

> restart:

According to S. L. Adler in *Quaternionic Quantum Mechanics and Quantum Fields* (Oxford University Press, 1995), a treatment of the canonical linear harmonic oscillator begins with a hamiltonian in this coordinate form for a particle of mass $\frac{1}{2}$ moving in one dimension,

$$H(x) = -i \left(\frac{\partial^2}{\partial x^2} \right) + a x^2$$

in which a denotes a quaternion imaginary constant parameter such that $a = -a^*$ and $a = i a_1 + j a_\beta$,

with a_β in a complex domain $[1, i]$. Splitting the assumed amplitude function ψ into components $\psi = \psi_\alpha + j \psi_\beta$, and substituting into the Schroedinger's temporally independent equation yields these two coupled complex equations:

$$\begin{aligned} \left(-i \left(\frac{\partial^2}{\partial x^2} \right) + i a_1 x^2 \right) \psi_\alpha - a_\beta^* x^2 \psi_\beta &= \psi_\alpha i E \\ \left(i \left(\frac{\partial^2}{\partial x^2} \right) - i a_1 x^2 \right) \psi_\beta + a_\beta^* x^2 \psi_\alpha &= \psi_\beta i E \end{aligned}$$

When a_β vanishes, taking $\psi_\beta = 0$ solves the second equation above; the first equation then reduces to the conventional equation for the harmonic oscillator in wave mechanics. When a_β is non-zero, solution of the former equation for ψ_β yields

$$\psi_\beta = \frac{1}{x^2 a_\beta^*} \left(-i \left(\frac{\partial^2}{\partial x^2} \right) + i a_1 x^2 - i E \right) \psi_\alpha$$

Substituting this result into the latter equation above yields, after some algebra, the following differential equation, of fourth order, for ψ_α .

$$\left(\left(- \left(\frac{\partial^2}{\partial x^2} \right) + a_1 x^2 \right)^2 - E^2 + \left(\frac{6}{x^2} - \frac{4 \left(\frac{\partial}{\partial x} \right)}{x} \right) \left(\frac{\partial^2}{\partial x^2} - a_1 x^2 + E \right) + |a_\beta|^2 x^2 \right) \psi_\alpha = 0$$

Solution of this equation to evaluate eigenfunction ψ_α and eigenvalue E enables one to evaluate ψ_β for the latter of the two coupled equations above.

```
> deq := (-diff(psi[alpha](x),x$2) + a[1]*x^2*psi[alpha](x))^2 -
E^2*psi[alpha](x) + (6/x^2*psi[alpha](x)-4*diff(psi[alpha](x),x)
/x)
```

```
*(diff(psi[alpha](x),x$2)-a[1]*x^2*psi[alpha](x)+E*psi
[alpha](x))+abs(a[beta])^2*x^2*psi[alpha](x) = 0;
```

$$deq := \left(- \frac{d^2}{dx^2} \psi_\alpha(x) + a_1 x^2 \psi_\alpha(x) \right)^2 - E^2 \psi_\alpha(x) + \left(\frac{6 \psi_\alpha(x)}{x^2} - \frac{4 \left(\frac{d}{dx} \psi_\alpha(x) \right)}{x} \right) \left(\frac{d^2}{dx^2} \right. \quad (23.1)$$

$$\left. \psi_\alpha(x) - a_1 x^2 \psi_\alpha(x) + E \psi_\alpha(x) \right) + |a_\beta|^2 x^2 \psi_\alpha(x) = 0$$

```
> assume(a[1]>0,E>0,a[beta]>0):
```

```
sol := dsolve(deq, psi[alpha](x));
```

```
sol := ( )
```

(23.2)

As only a trivial solution of this ordinary-differential equation appears, we proceed no further.

3.10 quantum mechanics in a chemical context

3.101 calculation of molecular electronic structure

We consider a molecular system of formula $C_2 H_6 O$ comprising 9 atomic nuclei (k, l) of the

specified elements plus 26 electrons (i, j); a molecule of this formula is assumed to possess electronic states of symmetry class ^{singlet}A so that there is no significant magnetic interaction apart from that due to intrinsic nuclear angular momenta that we neglect. In a general classical form, the hamiltonian for this system is a sum of kinetic energies of all particles and electrostatic potential energy between each two particles,

$$H(\underline{r}_i, \underline{R}_k) = T + V$$

$$= \sum_{i=1}^{26} \frac{\bar{p}_i^2}{2 m_e} + \sum_{k=1}^9 \frac{\bar{p}_k^2}{2 M_k} + \frac{e^2 \left(\sum_{i,j} \frac{1}{r_{ij}} + \sum_{k,l} \frac{Z_k Z_l}{R_{kl}} - \sum_{i,k} \frac{Z_k}{\rho_{ik}} \right)}{4 \pi \epsilon_o}$$

in which r_{ij} is a distance between two electrons, R_{kl} is a distance between two atomic nuclei and ρ_{ik} is a distance between electron i and nucleus k ; Z_k is the protonic number of nucleus k , \bar{p}_i and \bar{p}_k are operators to represent momenta of electrons and atomic nuclei respectively, and the three succeeding terms pertain to the electrostatic potential energy of repulsion between electrons and between atomic nuclei and of attraction between atomic nuclei and electrons; the latter sums run over all particles of the indicated type, electron or nucleus, but self interactions are excluded, i.e. when $i=j$ and $k=l$. Unlike the calculations in section groups 2.5 on the H atom and 3.6 on H₂⁺, we here include explicitly in our hamiltonian terms for the kinetic energy of translation of the atomic nuclei, relative to the centre of mass of the system; explicitly, we make no arbitrary separate treatment of electronic and nuclear motions but consider them jointly.

According to wave mechanics in the coordinate representation in which momentum p_j becomes a differential operator, $-\sqrt{-1} \frac{h}{2 \pi} \frac{\partial}{\partial q_j}$, for both electronic and nuclear particles for which $j = i$ or k respectively; under an assumption that the only interaction between the particles has the above electrostatic form, the complete hamiltonian for this system under these conditions becomes

$$H(\underline{r}_i, \underline{R}_k) = -\frac{h^2}{8 \pi^2 m_e} \sum_{i=1}^{26} \text{del}_i^2 - \frac{h^2}{8 \pi^2} \sum_{k=1}^9 \frac{\text{del}_k^2}{M_k}$$

$$+ \frac{e^2 \left(\sum_{i < j} \frac{1}{r_{ij}} + \sum_{k < l} \frac{Z_k Z_l}{R_{kl}} - \sum_{i,k} \frac{Z_k}{\rho_{ik}} \right)}{4 \pi \epsilon_o}$$

The first two terms, in which del_i^2 and del_k^2 denote the laplacian operator with second derivatives with respect to electronic and nuclear coordinates according to a chosen system of coordinates, pertain to the kinetic energy of electrons and atomic nuclei, respectively.

This hamiltonian operator $H(\underline{r}_i, \underline{R}_k)$ requires an operand for its differential quantities in laplacian operators del_i^2 and del_k^2 ; for this purpose we assume that $\Psi(\underline{r}_i, \underline{R}_k)$ is an amplitude function of both electronic and nuclear coordinates, such that both \underline{r}_i and \underline{R}_k are active variables. This notation is distinct from $\Psi(\underline{r}_i; \underline{R}_k)$ in which semicolon ; implies that the nuclear coordinates serve as mere parameters; as is shown below, the integrations involve both electronic and nuclear coordinates. Because for a system of multiple bodies no exact solution of Schroedinger's temporally independent equation

$$H(\underline{r}$$

$$\hat{H}(r_i, R_k) \Psi(r_i, R_k) = E \Psi(r_i, R_k)$$

seems to be practicable, after elimination of the motion of the centre of mass, we have recourse to integration, as follows,

$$E = \frac{\int \int \Psi(r_i, R_k) \hat{H}(r_i, R_k) \Psi(r_i, R_k) d\tau_i d\tau_k}{\int \int \Psi(r_i, R_k) \Psi(r_i, R_k) d\tau_i d\tau_k}$$

in which $d\tau_i$ and $d\tau_k$ are volume elements for integration over coordinates of all electrons and atomic nuclei respectively, and integrations are performed over all space, i.e. $-\infty \dots \infty$ with respect to cartesian coordinates.

Even after elimination of the motion of the centre of mass of this molecular system that yields a discrete state of least energy, the result of this integration is a sequence of values of energy, some discrete and some continuous, that apply to any molecule or molecules possessing in total a formula $C_2 H_6 O$, hence ethanol, or dimethyl ether, or methane plus methanal, or ethene plus water -- for that formula $C_2 H_6 O$ any combination of atomic nuclei and electrons that lacks significant magnetic interactions. As a result of this integration, the molecular system in a quantum state has indefinite extension in both space nor time -- no assignment of electrons to atomic nuclei, no atoms, no conformations, no isomers, no electric dipolar moment relative to coordinates in space: possessing spherical symmetry, such a molecular system acts like, so constitutes effectively, a point mass. This result is a particular example of a general conclusion by R. G. Woolley (*Advances in Physics*, 1976) that molecular structure, as a fairly rigid arrangement of atomic nuclei with their associated electrons, is incompatible with quantum mechanics.

Unlike the prototypical calculations on H_2^+ in section group 3.6 that yielded a curve of total energy as a function of distance R between the protons as atomic nuclei, the above calculation leaves no dependence on internuclear distances, merely yielding an energy, or sequence of energies, some discrete and some continuous; the reason for this distinct result is that in the calculations on H_2^+ the kinetic energy of the atomic nuclei was omitted -- the distance between those nuclei was fixed at R , and the energy of the system was calculated as a function of R . This procedure was first justified by M. Born and J. R. Oppenheimer in 1927, after the latter left Cambridge for Gottingen; much further consideration of this approximate treatment has resulted over the years; for an arrangement of atomic nuclei with their associated electrons that yields a local minimum of energy, this approximation makes possible an expansion in nuclear coordinates only in the immediate vicinity of the conformation that produces that energy minimum: there is no justification of a curve or hypersurface of potential energy between one energy minimum and another.

Likewise in the calculations in sections 3.82 and 3.83, an energy of a particular species was attainable as a function of internuclear distance, and possibly other properties of that species at some set of internuclear distances were also calculable. This condition is the practise of conventional quantum chemistry -- the calculation of various properties of chemical species, of which the most important might be the geometric parameters of the relative nuclear positions at a local or global minimum of electronic energy. Even though there is little explicit content of either theoretical chemistry or theoretical physics, such calculations on an empirical basis -- such and such a calculation performed in such and such a way yields some particular numerical value of some property -- provide meaningful chemical results, and are a worthy component of chemical investigation in company with experiment and true theory. The accuracy of any result, on comparison with experiment, owes more to the power of the algorithms and processors -- software and hardware -- than to any particular quantum-mechanical basis. Any such results comparable with an experimental quantity should, in no case, be associated with any artefact of a

particular algorithm, because, at least in principle, some other quantum-mechanical basis can provide the same result involving disparate artefacts. For the same reason, because the derivation above demonstrates clearly that quantum mechanics is incompatible with molecular structure, the major or only reason for chemists to study quantum mechanics, as a collection of algorithms, is to undertake, eventually, such calculations.

3.102 calculation on H_3^{+}

The calculation described in section 3.101 is at present hypothetical, because available computational resources are inadequate to complete the calculations of satisfactory accuracy within a practically acceptable duration, but calculations of only molecular electronic structure with fixed internuclear distances are entirely practicable, not only for that system comprising nine atomic nuclei and 26 electrons but also for systems of many more numerous particles. To undertake such calculations, internuclear distances in some particular set become an initial condition, and the result of the calculation corresponding to a local minimum of electronic energy reflects that initial condition. According to this separate treatment of electronic and nuclear motions, there is obtained at the end of the calculation, having attained a local minimum of electronic energy, internuclear separations and relevant angles that might be compared with an experimentally derived molecular structure, for instance from x-ray diffraction of a crystallographic sample or electron diffraction of a gaseous sample of a particularly selected chemical compound. Moreover, the slight variation of that electronic energy near the minimum as a function of selected internuclear distances might yield internal energies of which the differences are comparable with those of spectral transitions according to infrared or Raman spectra. The accuracy of such calculations, as a measure of the agreement with experimental spectral data, depends critically not only on the nature of the basis set of electronic amplitude functions or density functionals chosen to initiate the calculation of electronic energy but also on the extent of the account taken of electron correlation.

As an alternative approach without separation of electronic and nuclear motions, M. Cafiero and L. Adamowicz [*Chemical Physics Letters* 387 (2004) 136 -- 141] undertook an almost rigorous quantum-mechanical calculation on H_3^{+} , which molecule, as the simplest polyatomic species, contains three atomic nuclei, each with protonic number $Z = 1$, and two electrons, and which was discovered by J. J. Thomson in 1912; after a separation of variables pertaining to the centre of mass with an origin of coordinates set on one atomic nucleus, a solution of Schrodinger's equation in a non-relativistic approximation yields energies of internal, or spectrally pertinent, states of this polyatomic molecular ion. Quantum mechanics requires that all identical particles be treated as indistinguishable entities; because no operator representing distances between particles commutes with the coulombic hamiltonian, internuclear distances or *bond lengths* are immeasurable exactly for a polyatomic molecule in a stationary or quantum state. One might, however, calculate a distribution of probability -- an expectation value -- of a distance between particles of any two types, taking into account the indistinguishability of particles; for a diatomic molecule that contains only two atomic nuclei, the distinction between nucleus and electron enables a moderately accurate derivation of an average internuclear distance. Cafiero and Adamowicz reported *average distances* and their dispersions for the separations of nuclear particles in H_3^{+} , such as $R_{H-H} / 10^{-10} \text{ m} = 1.748 \pm 0.238$; that dispersion is larger than the classical vibrational amplitudes in the state of least energy. On that basis, it is impossible to conclude whether H_3^{+} is a *linear* or *triangular* molecule according to the geometric arrangement of the atomic nuclei; the average separation between atomic nucleus and electron, or between two

electrons, was not calculated. Applying isotopic substitution to undertake a calculation on HDT^{++} , so that these atomic nuclei become distinguishable, those authors concluded that the relative equilibrium nuclear geometry of that species has a shape of a nearly equilateral triangle.

Such a rigorous calculation on $^{12}\text{C}_6\text{H}_6$, in a quantum state would yield a single (mean) internuclear distance $\text{C-C} = 2.088 \cdot 10^{-10}$ m, a single (mean) internuclear distance $\text{C-H} = 2.688 \cdot 10^{-10}$ m, a single (mean) internuclear distance $\text{H-H} = 3.798 \cdot 10^{-10}$ m, and a single mean distance between electrons; there is no indication whatsoever from such a calculation of a molecular conformation or structure of benzene or other chemical species.

With regard to isomerism and chirality, B. T. Sutcliffe and R. G. Woolley [*Chemical Physics Letters* 408 (2005) 445-- 447] discussed classical molecular structure in relation to prospective quantum-mechanical calculations on HCN , C_3H_4 and C_8H_8 .

As a pragmatic and empirical procedure, separate and sequential treatment of electronic and nuclear motions might preserve a concept of molecular structure, but at a loss of quantum-mechanical justification of its application. Moreover, that procedure to calculate the electronic energy is valid only near a minimum of electronic energy -- between such multiple minima as might apply for a polyatomic molecule, there is no quantum-mechanical justification for the energy as a function of nuclear coordinates. Such concepts as *transition structures* have hence entirely no firm theoretical foundation, even though some particular computational procedure might produce such features of a hypersurface of potential energy of the atomic nuclei in a multidimensional space [B. T. Sutcliffe, R. G. Woolley, *Journal of Chemical Physics*, 137, 22A544, 2012].

3.103 significance for education in science

We consider further the fundamental postulate of quantum mechanics, which underpins all preceding calculations:

$$p_j q_k - q_k p_j = [p_j, q_k] = -\frac{i \hbar \delta_{j,k}}{2\pi},$$

in which p_j and q_k are components of conjugate variables momentum and position with assumed cartesian coordinates, respectively, and which becomes in one dimension simply

$$p q - q p = [p, q] = -\frac{i \hbar}{2\pi}.$$

What is the profound significance of this relation involving a non-zero commutator? On the left side of the equality appear products of formally purely real quantities -- a momentum and a position, whereas the right side is purely imaginary; this observation alone dictates that quantum mechanics is not a physical theory but a mathematical abstraction that might form a basis of algorithms for calculations; other quantities that arise within or from calculations based on this postulate are accordingly perceived to be possibly only mathematical artefacts of such calculations, not necessarily encompassing any physical or chemical reality.

According to matrix mechanics, each physical quantity is represented by a matrix, such as a matrix \mathbf{Q} for coordinate q , another matrix \mathbf{P} for coordinate p and a further matrix \mathbf{W} for energy W , as applied in section 1.22: although a matrix might in other circumstances act as an operator, for instance to operate on a vector either to produce an eigenvalue and the same vector or to rotate one vector into another vector, the matrix for energy in this calculation stands by itself; its non-zero elements only along the principal diagonal are, formally, physically observable quantities, or at least their differences would be

physically observable in practice through spectral transitions; the elements of the coordinate matrix are likewise directly, through their squares, related to probabilities of spectral transitions, or equivalently the measurable intensities of spectral lines. Although according to wave mechanics, in a coordinate or momentum representation, quantity either p or q becomes respectively a differential operator, the other quantity remains an algebraic quantity, not an operator other than a simple multiplicand, as applied in section 1.23. According to Dirac's approach, both p and q occur, by design, as a sum or difference of two operators -- one a^\dagger for creation and another a for destruction, as applied in section 1.24; although these quantities p and q might constitute operators in some contexts of these quantum-mechanical calculations, their dismissal as being operators invariably is thus inappropriate.

In considering these calculations in relation to a chemical context, one must bear in mind that various quantities or concepts might pertain to separate domains; the failure to recognise those domains produces a category fallacy [cf. H Primas, *Chemistry, Quantum Mechanics and Reductionism*, Springer Verlag, Berlin, Germany, 1983]. For instance, the experiments involving the measurement and manipulation of individual quantum systems, which warranted a Nobel prize in physics for 2012, belong to a physical domain -- that of quantum physics. The calculations undertaken in this worksheet belong to a mathematical domain; although such calculations might be applied to treat properties measured in those experiments, the domains of physics and mathematics to which each pertain remain distinct. We demonstrate in section 3.101 that a rigorous application of a quantum-mechanical calculation on a system of multiple atomic nuclei and multiple electrons in a state of discrete energy yields no conventional molecular structure. In contrast, we might imagine an hypothetical physical experiment in which two carbon atoms, six hydrogen atoms and one oxygen atom, i.e. conforming to a chemical formula C_2H_6O , might be placed in a small box; after some interval, spectrometric tests on the content of that box might reveal a molecule of ethanol, or of dimethyl ether, or even of methane and methanal or of ethene and H_2O , identified by lines in an absorption spectrum on comparison with spectra of authentic samples of these stable chemical compounds. In the latter experiment a molecular structure has *emerged*, in a sense that is impossible in the preceding calculation; if the spectral lines observed for identification involve rotational parameters, one might even eventually deduce a molecular structure from their analysis, according to classical moments of inertia of particular isotopic species. The contrast between the physical and mathematical domains is unambiguous.

With regard to the preceding calculations on the atomic system comprising one atomic nucleus and one electron, such as H in section group 2.5, and on a molecular system comprising two atomic nuclei and one electron, such as H_2^+ in section group 3.6, the primary results of those calculations, as of the preceding calculations on harmonic oscillators, are the energies of the systems. When wave mechanics is chosen as the quantum-mechanical algorithm, the results might include also the amplitude functions, variously denoted ψ , χ , ϕ ..., of spatial coordinates and perhaps also time, or alternatively amplitude functions of momenta. In the case of the H atom, these amplitude functions have been given a special name *orbital*, originated by R. S. Mulliken in 1932 and defined in his Nobel lecture with characteristic obfuscation as "something as much like an orbit as is possible in quantum mechanics". Such an orbital is precisely a mathematical function, a solution of a partial-differential equation, as derived in section 2.53a; in the case of the atomic system comprising one atomic nucleus and one electron as treated there, the amplitude function or amplitude function becomes an *atomic orbital*, whereas when with a single electron there are multiple atomic nuclei held in fixed relative positions during the calculation, as for H_2^+ in section group 3.6, the amplitude function or amplitude function becomes a *molecular orbital*; the nature of that molecular orbital of course varies according to the nature and conformation of the multiple atomic centres. Because these calculations take no account of repulsion between electrons -- necessarily, as the governing hamiltonian includes terms for only one

electron -- an atomic orbital is defined for only that H atom, or equivalent system, and a molecular orbital is defined for only a corresponding system with multiple atomic nuclei and one electron, such as, but not limited to, H_2^+ . An inherent property of these amplitude functions is their infinite nature, although there might exist lines or planes in which a particular function, in a non-relativistic approximation, has zero amplitude; an atom within a molecule is undefined -- a molecule comprises only atomic nuclei and their associated electrons. Any attempt to partition that electronic density is subject to arbitrary criteria. Although Bader launched a theory of atoms in molecules, that theory was based implicitly on densities of electronic charge in relation to a framework of atomic nuclei resulting from purported quantum-mechanical calculations, but such a molecular structure is incompatible with quantum mechanics, as explained above; electronic densities are measurable in principle in experiments with diffraction of x-rays from a small crystalline sample, but the results are subject to experimental error and to complication from the purported atomic motions at any temperature. For instance, a careful analysis of crystalline Li_2BeF_4 by x-ray diffraction according to both atomic and ionic models yielded nearly identical indices of the goodness of fit of 6345 independent reflexions of x-rays, but supplemental measurements on only 22 weak reflexions more sensitive to the electronic distribution indicated that the charge density far from the atomic centres is slightly better represented on a basis of neutral atoms than of ions [P. Seiler and J. Dunitz, *Helvetica Chimica Acta*, 69, 1107 - 1112, 1986]. Under such conditions, the claim to define atoms, rather than atomic centres, in crystalline samples is pure fantasy.

summary of quantum mechanics for chemistry

As quantum mechanics comprises a collection of mathematical procedures or algorithms applicable to calculations pertinent to systems on an atomic scale, and as many manifestations of discrete phenomena are an invaluable component of chemical education and chemical practice, an objective of this chapter is to create a mathematical foundation for the pursuit of such calculations. The topics include only a few of the types of mathematical procedures, listed in section 1.25, but the most important for chemical purposes. All algebraic, numerical and graphical capabilities of advanced mathematical software, such as *Maple*, are brought to bear on the development of the various topics and illustrations, just as for nearly any other part of physical science.

Content of this worksheet is copyright 2021 J. F. Ogilvie and contributors.