Katholieke *Universiteit* Nijmegen

Quasi-adiabatic method to solve the coupled channels problem

Hoofdvak verslag Informatische Chemie

M.C.G.N. Van Vroonhoven

April 16, 1998

Institute of Theoretical Chemistry
Toernooiveld 1
Nijmegen

Contents

Samenvatting

Om op moleculair niveau de reactie A + BC → AB + C te begrijpen, moet quantummechanisch het probleem van de reactieve botsing van A met BC bestudeerd worden. De quantummechanica geeft (in principe) de waarschijnlijkheid dat de reactie plaats vindt, als functie van de relatieve kinetische energie van A en BC en de toestand van BC. Om deze waarschijnlijkheid echt uit te rekenen zijn zeer geavanceerde computermethodes vereist. Het computerprogramma DYNA[1] is in Nijmegen voor dit doel ontwikkeld. Dit hoofdvakverslag beschrijft een tweetal optimalisaties van een stukje van DYNA. Dit werk werd uitgevoerd op de afdeling Theoretische Chemie van de katholieke universiteit van Nijmegen, in het kader van een hoofdvakstage Informatische Chemie.

De tijdsonafhankelijke Schrödingervergelijking wordt omgeschreven naar een matrix vergelijking, de gekoppelde-kanalenvergelijking. Gekoppelde kanalen kunnen gebruikt worden om inelastische verstrooiïngsproblemen op te lossen, of om distorted waves (verstoorde golven) te construeren. Distorted waves kunnen als basisfunctie gebruikt worden in een variationele reactieve verstrooiïngsberekening. Ze worden dan gebruikt om de golffunctie in de zogenaamde buitengebieden van de coordinaatruimte te beschrijven. De buitengebieden zijn die stukken van de coordinaatruimte waarin geen reactieve interactie op kan treden tussen de twee deeltjes. In deze gebieden kan het probleem dus met inelastische verstrooiïngstheorie beschreven worden. Het uitrekenen van de distorted waves gebeurt door de gekoppelde-kanalenvergelijking op te lossen.

De optimalisatie bestaat uit twee stappen. De eerste stap is numeriek wiskundig van aard. De gekoppelde-kanalenvergelijking is een tweede orde differentiaalvergelijking in de afstand R tussen A en BC, die met bepaalde probleemafhankelijke randvoorwaarden opgelost moet worden. Deze vergelijking wordt opgelost door propagatie van de oplossing van de ene rand van het interval naar de andere. Daartoe wordt de differentiaal vergelijking omgeschreven naar een serie beginvoorwaardenproblemen over aangrenzende stukken van het totale propagatie interval. Deze beginvoorwaardenproblemen worden dan opgelost met behulp van een standaard routine uit de NAG software bibliotheek. De oplossingen van deze beginvoorwaardenproblemen worden dan omgeschreven naar de oplossingen van de randvoorwaarden-problemen over de betreffende stukken van het totale propagatie interval. Deze oplossingen worden dan aan elkaar geplakt, wat resulteert in de totale oplossing van het oorspronkelijke randvoorwaardenprobleem.

De eerste stap bestaat uit het optimaliseren van het gebruik van deze bibliotheek routine. De routine heeft een aantal parameters om de fout in het antwoord te controleren. Er zijn tests uitgevoerd aan het model probleem $y'' = k^2y$. Uit de resultaten van deze tests kunnen twee formules afgeleid worden, waarmee de optimale parameter keuzes voor ons type problemen berekend kunnen worden.

Uit test berekeningen aan de inelastische botsing van H met H2 blijkt dat deze optimalisatie de berkeningen circa 30 mogelijk om de gekoppelde-kanalenvergelijking met een vooraf bepaalde nauwkeurigheid op te lossen.

De tweede optimalisatie is van theoretisch chemische aard. Er is een nieuwe methode ontwikkeld om de gekoppelde-kanalenvergelijking op te lossen. Deze nieuwe methode werkt voor willekeurige randvoorwaarden, dus zowel voor inelastische problemen als voor de buitengebieden van een reactief probleem. Vanuit de literatuur zijn er twee typen methoden bekend om de gekoppelde-kanalenvergelijking met inelastische randvoorwaarden op te lossen. De twee typen verschillen in het soort basisfuncties dat gebruikt wordt, welke benaderingen er gemaakt worden om de vergelijking op te lossen, en hoeveel verschillende sets basisfuncties er gebruikt worden.

Het eerste type methoden heet "adiabatisch", waar "langzaam veranderend" mee bedoeld wordt. Bij dit type methoden wordt aangenomen dat de interactie potentiaal tussen de twee botsende deeltjes langzaam verandert met de afstand tussen de deeltjes. Dit houdt in dat er verondersteld wordt dat deze interactie potentiaal constant is over een klein stukje van het propagatie interval. Nu wordt de potentiaal matrix gediagonaliseerd. Dit geeft een basis van locale eigenfuncties van de Hamiltoniaan. De gekoppelde-kanalenvergelijking is voor een probleem met een constante diagonale potentiaal exact oplosbaar. De benaderde oplossing over het kleine stukje van het propagatie interval wordt nu gegeven door de exacte oplossing voor de benaderde constante potentiaal. Dit wordt herhaald, tot over het hele interval oplossingen bekend zijn. Deze stukjes oplossing worden dan aan elkaar geplakt tot de totale oplossing.

Het tweede type methoden heet "diabatisch", het tegenovergestelde van adiabatisch. Hier wordt niet de aanname gemaakt dat de potentiaal constant verondersteld kan worden over een klein stukje interval. De gekoppelde-kanalenvergelijking wordt numeriek opgelost, de potentiaal wordt niet benaderd. Als basisset worden de eigenfuncties gebruikt voor het systeem waarin de botsende deeltjes zich zeer ver van elkaar af bevinden, en dus geen interactie meer hebben.

Beide methoden hebben voor- en nadelen. Het nadeel van het eerste type methoden is, dat het stukje waarover de potentiaal constant verondersteld kan worden, erg klein is. Dit houdt in dat de potentiaal matrix erg vaak gediagonaliseerd moet worden, wat veel computertijd kost. Het voordeel is dat er gewerkt wordt in een basis van eigenfuncties, die het systeem goed beschrijven. Er zijn dus maar weinig basisfuncties nodig, waardoor de dimensie van de matrices niet erg groot wordt. Het nadeel van het tweede type methoden is, dat de basisfuncties geen eigenfuncties van het systeem zijn, er zelfs helemaal niet op lijken. Daarom zijn er veel basisfuncties nodig om het systeem te beschrijven. Het voordeel is, dat er maar een basis gebruikt wordt, en dat er dus niet getransformeerd hoeft te worden.

De nieuwe methode probeert de voordelen van de beide methoden te behouden, en de nadelen te vermijden. Het idee is om het totale propagatie interval in een aantal stukken te verdelen, waarin dan een locaal optimale basis berekend wordt. De gekoppelde-kanalenvergelijking wordt dan numeriek opgelost in deze locale basis. Dit heeft als voordeel, dat de basisfuncties het systeem goed beschrijven, er zijn dus minder basisfuncties nodig dan bij een diabatische methode. Omdat de gekoppelde-kanalenvergelijking wel numeriek gepropageerd wordt, kan het interval waarover de locale basis gebruikt wordt, veel groter zijn dan bij de adiabatische methoden. Er zijn dus veel minder basistransformaties nodig. Omdat de methode een combinatie is van de oorspronkelijke twee typen, wordt hij "quasi-adiabatisch" genoemd. Verder heeft de quasi-adiabatische methode als voordeel dat van te voren opgegeven kan worden hoe nauwkeurig ht eindresultaat moet zijn. Hiervoor is slechts een minimaal aantal parameters nodig, die zoveel mogelijk een fysische betekenis hebben. Alle overige parameters worden automatisch berekend. Dit maakt het mogelijk dat de routine als een blackbox achtige routine geimplementeerd kan worden, wat ook gedaan is in het computerprogramma pakket DYNA.

Uit test berekeningen aan het reactieve systeem H + O_2, blijkt dat de quasi-adiabatische methode die in dit werk ontwikkeld is vooral erg efficient is, als er resultaten met een hoge nauwkeurigheid vereist zijn. Dan is de methode ongeveer twee keer zo snel als de oude, diabatische methode. Samen met de eerste stap van de optimalisatie geeft dit een computerprogramma dat ongeveer een factor drie sneller is dan tevoren.

Chapter 1

Introduction

This report is about the work the author performed as an undergraduate student in computational chemistry ("Informatische Chemie") at the Department of Theoretical Chemistry of the University of Nijmegen. The assignment was the optimization of part of an existing computer code for atom-diatom reactive scattering, by developing a new method for the inelastic part of the problem.

The aim of a (reactive or inelastic) scattering calculation is to obtain probabilities for transitions from a particular initial state to another (final) state under influence of the collision. These transition probabilities can be calculated from the scattering matrix ($\mathbb{S}$). This matrix can be calculated from the wave function of the system. Therefore the Schrödinger equation must be solved.

We use internal Jacobi coordinates to describe our system. These coordinates are shown in figure 1.1 for the A + BC arrangement. These

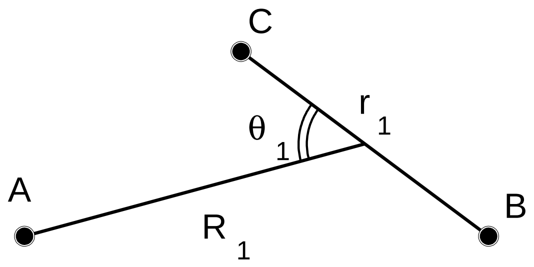

Figure 1.1: Coordinate system for an atom-diatom collision in arrangement 1

coordinates describe only the internal configuration of the system. We consider only systems with no total angular momentum ($J = 0$), and without external fields. Therefore, the position of the center of mass (CM) and the overall orientation of the system in space do not need to be considered. An arrangement means a constellation of the three atoms, in which some of the atoms may be chemically bound to other atoms. With three different atoms A, B, and C the following three arrangements are important:

- The A + BC arrangement, with molecule BC and atom A.
- The B + AC arrangement, with molecule AC and atom B.
- The C + AB arrangement, with molecule AB and atom C.

When two of the three atoms are the same (atoms A, A and B), the two arrangements in which atom B is bounded to an atom A are the same. There remain only two arrangements, A + AB and B + AA. The total energy (E) is chosen so that the arrangement A + B + C does not occur. We also assume that the ABC is not bound.

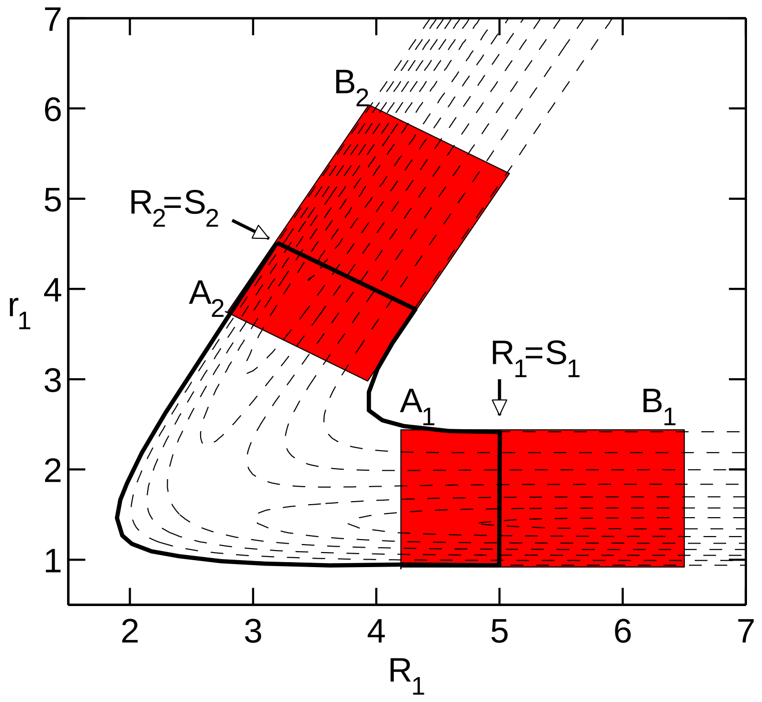

Figure 1.2: Area surrounded by solid line: interaction region. Grey parts: external regions. Note that the interaction and external region overlap.

The total coordinate space is divided in an internal (or interaction) region and the external regions. There are three external regions, one for each arrangement. This is visualized in figure 1.2, which gives a contour plot of the potential energy surface for the special case of the collinear collision of an atom A and a molecule BC. In this collinear collision only two arrangements are important, A + BC and AB + C. The coordinate R_1 describes the distance between A and the center of mass of BC, and r_1 describes the BC distance. If all masses are equal it can be easily shown that the contours in the second arrangement have asymptotically the slope two.

The borders between the regions are defined, such that in the internal region reaction can occur, and in the external regions only elastic and inelastic events may occur. The region outside the external region ($R > B$) may be called the asymptotic region, in this region there is no interaction at all between the fragments.

In the external regions, distorted waves (DWs) are constructed. These DWs are solutions to the Schrödinger equation (SE). There are three types of DWs, regular, irregular, and closed. A closed DW is an asymptotically vanishing solution of the SE. Regular and irregular DWs do not vanish asymptotically. Together, they form the set of open DWs. The difference between these two types lies at the origin where regular DWs go to zero, and irregular DWs do not vanish. The regular DWs are the physical solutions to purely inelastic problems. In solving a reactive problem, also the irregular and closed DWs are needed. In the asymptotic region, there is no interaction between the fragments. The solutions to the SE are then given by the free waves. Since there is interaction in the external region, the solutions are no longer free waves. The solutions are still wave-like, they are called "distorted waves".

In an (in)elastic scattering problem, there is no interaction region and the DWs are the solutions of the radial part of the problem. In a reactive problem, the DWs serve as basis functions in a variational method to solve the total reactive problem. For this reason the interaction and external regions overlap.

In this report the focus is on the external regions. The aim of the project is to optimize the construction of the distorted waves. To construct them, the SE equation has to be solved. To that end it is rewritten as a matrix differential equation, the coupled channels (CC) equation. This equation is then solved with an algorithm that uses a differential equation solving routine from the NAG library.[2]

The outline of this report is as follows. In chapter 2 the derivation of the CC equation is given. In chapter 3 follows a discussion on the algorithm, used to solve the CC equation. In this algorithm, a routine from the NAG library is used. As a first optimization, the use of this routine was studied. It appears that efficient usage of this routine may speed up the calculation very much. This optimization is reported in chapter 4.

The DWs are required to be good solutions of the SE. Therefore we need to include classically forbidden basis functions (closed channels) in our calculation. These closed channels give rise to exponentially growing and decaying solutions as a function of R. With these kind of solutions, it is difficult to calculate all the linear independent solutions over a large interval. Several methods are known in literature[3–7] to solve the problems arising from propagating the CC equations over larger intervals.

We propose a new method with several advantages over the other methods. It can be used to construct regular, irregular and closed DWs. All the parameters in the problem, are determined automatically as much as possible. The step size and the number of basis functions are variable over the interval. Towards the end of the integration interval, where the interaction potential between the atom and the molecule goes to zero, the method will use fewer functions, and will make larger steps, because the problem is easier. Furthermore the method uses a different, locally optimal basis at each step. This makes it possible to develop a black box like routine to solve the CC equations in a fast way. The results and comparison with other methods will be presented in chapter 5.

In the last chapter, all conclusions will be summarized, and some remarks on future research will be made.

Chapter 2

The coupled channels equation for atom-diatom scattering

In this chapter, a derivation of the coupled channels (CC) equation for atom-diatom scattering is given. This equation is a special matrix form of the Schrödinger equation (SE). This equation is solved to compute distorted waves, the solutions to the SE in the external regions [Fig. 1.2].

The derivation is given in a few steps. In the first section, the Schrödinger equation in a suitable coordinate system is derived for collinear inelastic collisions. In section 2.2, the CC equation for this type of collisions is derived from this SE. In section 2.3, these results are generalized to non-collinear inelastic atom-diatom collisions with zero total angular momentum ($J = 0$). After that, in section 2.4 a discussion on the initial conditions, needed to solve the differential equations, is given. In the last section a small, analytically solvable, example is given. From this example, the problems encountered in propagating the CC equations become clear.

See Secrest and Johnson[8] for a detailed discussion of collinear atom-diatom collisions.

2.1 Schrödinger equation for a collinear collision

Consider a collision of an atom (particle A) and a diatomic molecule BC. If the masses of the particles are given by m_A, m_B, and m_C, and the Cartesian positions by x_A, x_B, and x_C, the SE for this system is

$$-\frac{\hbar^2}{2m_A}\frac{\partial^2\Psi}{\partial x_A^2} - \frac{\hbar^2}{2m_B}\frac{\partial^2\Psi}{\partial x_B^2} - \frac{\hbar^2}{2m_C}\frac{\partial^2\Psi}{\partial x_C^2} + V(x_A, x_B, x_C)\Psi = E\Psi. \qquad (2.1)$$

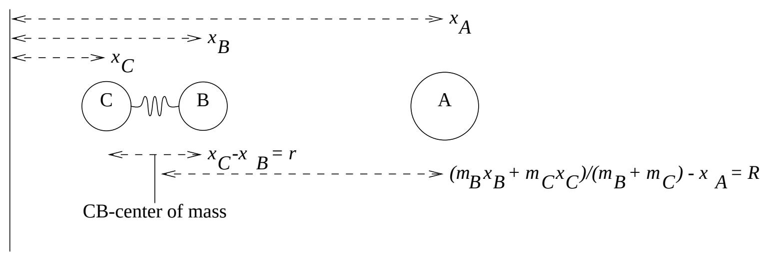

Figure 2.1: Coordinate system for the collinear collision of an atom and a diatomic molecule

The Cartesian coordinate system is not very practical; in calculations one rather uses the center of mass (CM) coordinates:

$$\begin{aligned} r &= x_C - x_B \\ R &= \frac{m_B x_B + m_C x_C}{m_B + m_C} - x_A \\ X &= \frac{m_A x_A + m_B x_B + m_C x_C}{m_A + m_B + m_C}. \end{aligned} \tag{2.2}$$

where r is the intermolecular distance, R is the distance between the atom A and the CM of the molecule BC, and X the position of the CM of the whole system. The volume element does not change in this change of coordinate system, since the determinant of the Jacobian between these two coordinate systems

$$\begin{bmatrix} \frac{\partial r}{\partial x_A} & \frac{\partial r}{\partial x_B} & \frac{\partial r}{\partial x_C} \\ \frac{\partial R}{\partial x_A} & \frac{\partial R}{\partial x_B} & \frac{\partial R}{\partial x_C} \\ \frac{\partial X}{\partial x_A} & \frac{\partial X}{\partial x_B} & \frac{\partial X}{\partial x_C} \end{bmatrix} = \begin{bmatrix} 0 & -1 & 1 \\ -1 & \frac{m_B}{m_B+m_C} & \frac{m_C}{m_B+m_C} \\ \frac{m_A}{M} & \frac{m_B}{M} & \frac{m_C}{M} \end{bmatrix} \tag{2.3}$$

equals unity. In this Jacobian, $M = m_A + m_B + m_C$. Using this, one finds

$$-\frac{\hbar^2}{2m_A}\frac{\partial^2}{\partial x_A^2} = -\frac{\hbar^2 m_A}{2M^2}\frac{\partial^2}{\partial X^2} + \frac{\hbar^2}{M}\frac{\partial^2}{\partial X \partial R} - \frac{\hbar^2}{2m_A}\frac{\partial^2}{\partial R^2}, \tag{2.4}$$

$$\begin{aligned} -\frac{\hbar^2}{2m_B}\frac{\partial^2}{\partial x_B^2} &= \\ &-\frac{\hbar^2 m_B}{2M^2}\frac{\partial^2}{\partial X^2} - \frac{\hbar^2 m_B}{M(m_B + m_C)}\frac{\partial^2}{\partial X \partial R} + \frac{\hbar^2}{M}\frac{\partial^2}{\partial X \partial r} \\ &-\frac{\hbar^2 m_B}{2(m_B + m_C)^2}\frac{\partial^2}{\partial R^2} + \frac{\hbar^2}{m_B + m_C}\frac{\partial^2}{\partial R \partial r} - \frac{\hbar^2}{2m_B}\frac{\partial^2}{\partial r^2}, \end{aligned} \tag{2.5}$$

and

$$
\begin{aligned}
-\frac{\hbar^2}{2m_C}\frac{\partial^2}{\partial x_C^2} = \\
& -\frac{\hbar^2 m_C}{2M^2}\frac{\partial^2}{\partial X^2} - \frac{\hbar^2 m_C}{M(m_B+m_C)}\frac{\partial^2}{\partial X \partial R} - \frac{\hbar^2}{M}\frac{\partial^2}{\partial X \partial r} \\
& -\frac{\hbar^2 m_C}{2(m_B+m_C)^2}\frac{\partial^2}{\partial R^2} - \frac{\hbar^2}{m_B+m_C}\frac{\partial^2}{\partial R \partial r} - \frac{\hbar^2}{2m_C}\frac{\partial^2}{\partial r^2}.
\end{aligned} \tag{2.6}
$$

The potential formally becomes a function of the three new coordinates. Because there is no external field applied, the potential will not depend on the position of the the CM, so the potential term becomes $V(r, R)$. Substituting the previous results into the Schrödinger equation [Eq. (2.1)] gives:

$$
-\frac{\hbar^2}{2M}\frac{\partial^2 \Psi}{\partial X^2} - \frac{\hbar^2}{2\mu}\frac{\partial^2 \Psi}{\partial R^2} - \frac{\hbar^2}{2m}\frac{\partial^2 \Psi}{\partial r^2} + V(r,R)\Psi = E\Psi, \tag{2.7}
$$

where the the scattering mass of the system (the reduced mass of the atom A with respect to the molecule BC, μ), and the vibrational mass (the reduced mass of the molecule BC, m) are given by

$$
\begin{aligned}
\mu &= \frac{m_A(m_B+m_C)}{M}, \\
m &= \frac{m_B m_C}{m_B+m_C}.
\end{aligned}
$$

As stated before, M is the total mass of the system.

The first term of the kinetic energy represents the motion of the CM. The second term describes the motion along the scattering coordinate, i.e. the motion of atom A towards and away from the molecule BC. The last term of the kinetic energy describes the internal vibrational motion of the molecule BC.

2.2 The coupled channels equation

It depends on the interaction potential $V(r, R)$ whether the SE [Eq. (2.7)] can be solved analytically. In most cases this will not be possible and it has to be solved numerically. In this section, the SE is rewritten to a matrix equation, that can be solved on a computer. Of course, the matrix is of finite dimension N and hence an approximation is introduces. By varying N one can stude the size of the error made by this approximation.

One can define the potential for the atom and the molecule at large separation:

$$
V_0(r) \equiv \lim_{R\to\infty} V(r,R). \tag{2.8}
$$

With this, one can rewrite

$$V(r,R) = V_0(r) + \Delta V(r,R), \tag{2.9}$$

where $\Delta V(r,R) = V(r,R) - V_0(r)$. Assume that the SE for the molecule BC (without atom A) has been solved:

$$\left[-\frac{\hbar^2}{2m}\frac{\partial^2}{\partial r^2} + V_0(r)\right]\varphi_i(r) = \varepsilon_i \varphi_i(r). \tag{2.10}$$

Assuming the solutions $\varphi_i, i = 1, 2, \ldots$ to form a complete orthonormal set, we can expand the wave functions in this basis:

$$\Psi_j(r,R) = \sum_{i=1}^{N} u_{ij}(R)\varphi_i(r), \tag{2.11}$$

where Ψ_j is the j-th linearly independent solution to the SE [Eq. (2.7)], and u_{ij} are the expansion coefficients, which have to be determined. The number of basis functions (N) is used as a convergence parameter.

Substituting Eqs. (2.9), (2.10), and (2.11) into the SE gives:

$$\begin{aligned}\sum_{i=1}^{N}\left[-\frac{\hbar^2}{2\mu}\frac{d^2u_{ij}(R)}{dR^2}\varphi_i(r) + u_{ij}(R)\varepsilon_i\varphi_i(r) + u_{ij}(R)\Delta V(r,R)\varphi_i(r)\right]& \\ = E\sum_{i=1}^{N}u_{ij}(R)\varphi_i(r).&\end{aligned} \tag{2.12}$$

Multiplying this equation by $\varphi_k^*(r)$ and integrating over r gives:

$$\begin{aligned}-\frac{\hbar^2}{2\mu}\frac{d^2u_{kj}(R)}{dR^2} + u_{kj}(R)(\varepsilon_k - E) +& \\ \sum_{i=1}^{N}u_{ij}(R)\int\varphi_k^*\Delta V(r,R)\varphi_i(r)dr = 0.&\end{aligned} \tag{2.13}$$

This equation is valid for all k between 1 and N. Defining the matrices

$$\mathbb{U}(R) \equiv \begin{bmatrix} u_{11}(R) & u_{12}(R) & \cdots & u_{1N}(R) \\ u_{21}(R) & u_{22}(R) & \cdots & u_{2N}(R) \\ \vdots & \vdots & \ddots & \vdots \\ u_{N1}(R) & u_{N2}(R) & \cdots & u_{NN}(R) \end{bmatrix}, \tag{2.14}$$

$$\mathbb{E} \equiv \begin{bmatrix} E-\varepsilon_1 & 0 & \cdots & 0 \\ 0 & E-\varepsilon_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & E-\varepsilon_N \end{bmatrix}, \tag{2.15}$$

and $\mathbb{V}(R)$ with elements

$$\mathbb{V}_{ij}(R) \equiv \int \phi_i^*(r)\Delta V(r,R)\phi_j(r)dr \tag{2.16}$$

transforms the SE into the following matrix differential equation

$$\mathbb{U}''(R) - \frac{2\mu}{\hbar^2}[\mathbb{V}(R) - \mathbb{E}]\mathbb{U}(R) = 0. \tag{2.17}$$

Now define the $\mathbb{W}$ matrix:

$$\mathbb{W}(R) = \frac{2\mu}{\hbar^2}[\mathbb{V}(R) - \mathbb{E}]. \tag{2.18}$$

Substituting this in the differential equation yields the coupled channels equation

$$\mathbb{U}''(R) = \mathbb{W}(R)\mathbb{U}(R). \tag{2.19}$$

2.3 The CC equation for a three dimensional system

The coordinate system for a general atom-diatom collision was defined in figure 1.1. The vector $\vec{r}$ with length r points from atom B to atom C, and $\vec{R}$ (with length R) points from the CM of BC to atom A. In space-fixed, cartesian coordinates the Hamiltonian is given by

$$\hat{H} = -\frac{\hbar^2}{2\mu}\nabla_R^2 - \frac{\hbar^2}{2m}\nabla_r^2 + V_0(r) + \Delta V\left[R, r, \vec{R}\cdot\vec{r}/(Rr)\right] - \frac{\hbar^2}{2M}\nabla_X^2. \tag{2.20}$$

The center of mass kinetic energy, $-\frac{\hbar^2}{2M}\nabla_X^2$, can be ignored. This Hamiltonian can be derived in the same way as Eq. (2.7). The interaction potential vanishes for large R:

$$\lim_{R\to\infty} \Delta V\left[R, r, \vec{R}\cdot\vec{r}/(Rr)\right] = 0. \tag{2.21}$$

The full potential is again given by

$$V(R, r, \vec{R}\cdot\vec{r}) = V_0(r) + \Delta V\left[R, r, \vec{R}\cdot\vec{r}/(Rr)\right], \tag{2.22}$$

where $V_0(r)$ is the potential for the free diatom BC. The reduced masses m and μ and the total mass M are the same as in the collinear case.

In spherical polar coordinates we have

$$\nabla_R^2 = \frac{1}{R^2}\left[\frac{\partial}{\partial R}R^2\frac{\partial}{\partial R} + \hat{l}^2\right] \tag{2.23}$$

$$\nabla_r^2 = \frac{1}{r^2}\left[\frac{\partial}{\partial r}r^2\frac{\partial}{\partial r} + \hat{j}^2\right]. \tag{2.24}$$

The polar angles of the vectors $\vec{R}$ and $\vec{r}$ are given by $\hat{R}$ and $\hat{r}$. In the asymptotic region, separation of $\vec{R}$ and $\vec{r}$ is possible, and the wave function is of the product form:

$$\Phi(R)\left[\chi_i^{\text{vib}}(r)/r\right] Y_{l,m_l}(\hat{R}) Y_{j,m_j}(\hat{r}), \tag{2.25}$$

where $i \equiv (j, v)$ is a label running over all rovibrational states of the diatom BC. In the external region, the two angular momenta couple to a total angular momentum $\vec{J} = \vec{j} + \vec{l}$. The angular part of the wave function is given by the Clebsch-Gordan coupling

$$\left[Y_l(\hat{R}) \times Y_j(\hat{r})\right]_M^J. \tag{2.26}$$

In this report, only the special case $J = 0$ is considered. In this case the wave function is rotational invariant. Rotate the system such that $\hat{R}$ is along the space-fixed z-axis. Then ($j = l$):

$$\begin{aligned} \left[Y_l(\hat{R}) \times Y_j(\hat{r})\right]_0^0 &= \sum_m (-1)^m Y_{j,m}(0,0) Y_{j,-m}(\hat{r}) \\ &= \sqrt{\frac{2j+1}{4\pi}} Y_{j,0}(\hat{r}) \\ &= P_j(\cos\theta), \end{aligned} \tag{2.27}$$

where the spherical polar angle θ is given by $\hat{r} \cdot \hat{R} = \cos\theta$. Hence the asymptotic wave function is given by

$$\Phi(R)\left[\chi_i^{\text{vib}}(r)/r\right] P_j(\cos\theta). \tag{2.28}$$

The channel eigenfunctions

$$\varphi_i(r, \cos\theta) = P_j(\cos\theta)\chi_i^{\text{vib}}(r)/r \tag{2.29}$$

are eigenfunctions of the diatomic Hamiltonian, with eigenvalues ε_i:

$$\left[-\frac{\hbar^2}{2m}\nabla_r^2 + V_0(r) - \varepsilon_i\right] \varphi_i(r, \cos\theta) = 0. \tag{2.30}$$

The total wave function for any R can now be expanded in terms of these channel eigenfunctions:

$$\Psi_j(r, R, \theta) = \sum_i u_{ij}(R)\varphi_i(r, \cos\theta)/R. \tag{2.31}$$

Substitution in the SE, and multiplying by $\varphi_k^*(r, \cos\theta)$ and integrating over r and θ (with the correct volume element $d\tau = r^2 \sin\theta dr d\theta$) gives again (just as in the two dimensional case)

$$\frac{d^2}{dR^2}u_{kj}(R) - \frac{j(j+1)}{R^2}u_{kj}(R) - \frac{2\mu}{\hbar^2}(\varepsilon_k - E)u_{kj}(R) \\ - \frac{2\mu}{\hbar^2}\sum_i u_{ij}(R)\int \varphi_k^*(r,\cos\theta)\Delta V(R,r,\cos\theta)\varphi_i(r,\cos\theta)d\tau = 0. \quad (2.32)$$

When the matrices $\mathbb{U}(R)$ and $\mathbb{E}$ are defined as in Eqs. (2.14) and (2.15), and we define the $\mathbb{V}$ matrix as

$$\mathbb{V}_{ki}(R) = \int \varphi_k^*(r,\cos\theta)\Delta V(R,r,\cos\theta)\varphi_i(r,\cos\theta)d\tau, \quad (2.33)$$

we get for the SE:

$$\mathbb{U}''(R) = \mathbb{W}(R)\mathbb{U}(R), \quad (2.34)$$

with the new definition for the $\mathbb{W}$ matrix

$$\mathbb{W}_{ki}(R) = \left[\frac{j(j+1)}{R^2} - \frac{2\mu}{\hbar^2}\mathbb{E}_{ki}\right]\delta_{ki} + \frac{2\mu}{\hbar^2}\mathbb{V}_{ki}(R), \quad (2.35)$$

where j is the rotational quantum number associated with channel eigenfunction φ_k.

2.4 Boundary conditions

In the previous section a second order matrix differential equation has been derived. If the boundary conditions on the wave function or on the expansion coefficients $\mathbb{U}$ are given, this equation can be solved numerically.

2.4.1 Inelastic scattering

In a purely inelastic problem, it is demanded that the wave function is regular at the origin. This gives the first boundary condition

$$\mathbb{U}(0) = \mathbf{0}. \quad (2.36)$$

In the asymptotic region, the wave function must behave like an incoming plane wave in the entrance channel, plus outgoing spherical waves in all the channels. The amplitude on each of the outgoing waves is given by the scattering matrix ($\mathbb{S}$). In the asymptotic region, $\mathbb{W}$ is diagonal and constant. In section 2.6 it is shown that the CC equation can be solved analytically for a constant $\mathbb{W}$ matrix.

We define $\mathbb{F}$ to be the set of regular solutions in the asymptotic region:

$$\mathbb{F}_{ij}(R) = \delta_{ij}\sqrt{\frac{k_i}{\hbar\mu}}\sin(k_i R), \quad (2.37)$$

and $\mathbb{G}$ to be the set of corresponding irregular solutions:

$$\mathbb{G}_{ij}(R) = -\delta_{ij}\sqrt{\frac{k_i}{\hbar\mu}}\cos(k_i R), \tag{2.38}$$

where k_i is the asymptotic wavenumber of the i-th channel:

$$k_i = \sqrt{-\mathbb{W}_{ii}(R \to \infty)} = \sqrt{\frac{2\mu(E-\varepsilon_i)}{\hbar^2}} \tag{2.39}$$

Incoming and outgoing solutions can be defined as linear combinations of these two sets of solutions:

$$\mathbb{U}_{\text{in}}(R) = \mathbb{G}(R) + i\mathbb{F}(R) \tag{2.40}$$
$$\mathbb{U}_{\text{out}}(R) = -\mathbb{G}(R) + i\mathbb{F}(R) \tag{2.41}$$

This gives the second boundary condition on the expansion coefficients, at the matching point $R = B$ between the external and asymptotic region:

$$\mathbb{U}(B) = \mathbb{U}_{\text{in}}(B) + \mathbb{U}_{\text{out}}(B)\mathbb{S} \tag{2.42}$$

2.4.2 External region in reactive scattering

For reactive scattering, the physical boundary conditions are given by the demand that the wave function is regular at the origin again. The asymptotic behavior should be an incoming wave in one channel of one arrangement, and outgoing waves in all the other channels (in all the arrangements). If all the channels from the different arrangements are put into one matrix, the boundary conditions look exactly like Eqs. (2.36) and (2.42).

The reactive $\mathbb{S}$ matrix follows from the variational calculation, in which the distorted waves are used as basis functions. The optimal boundary conditions for the DWs follow from numerical considerations.[9] These optimal boundary conditions can only be computed if a full set of DWs has already been computed. Therefore we use as boundary conditions for the expansion coefficients

$$\mathbb{U}(A) = \mathbf{0} \text{ and } \mathbb{U}(B) = \mathbf{1} \tag{2.43}$$

for the regular DWs, and

$$\mathbb{U}(A) = \mathbf{1} \text{ and } \mathbb{U}(B) = \mathbf{0} \tag{2.44}$$

for the irregular DWs. These boundary conditions give two sets of solutions. The solutions that obey the optimal boundary conditions, can then be calculated as linear combinations of the functions of these sets.

It is also possible to construct an inelastic $\mathbb{S}$ matrix from the calculated regular DWs. This has been used as a test for the algorithm presented in chapter 5. The quality of this constructed inelastic $\mathbb{S}$ matrix is a good measure for the quality of the DWs, and using this test avoids the variational calculation of a reactive $\mathbb{S}$ matrix each time, which is very time consuming. This inelastic matrix can be calculated by matching the regular DWs to the second boundary condition [Eq. (2.42)].

2.5 Properties of the scattering matrix

In this section, some important properties of the $\mathbb{S}$ matrix are listed. The most important property, is that the transition probabilities follow from the scattering matrix. The probability on scattering from channel i into channel j ($\mathbb{P}_{ij}$) is given by the square of the corresponding $\mathbb{S}$ matrix element:

$$\mathbb{P}_{ij} = |\mathbb{S}_{ij}|^2 \tag{2.45}$$

Furthermore the $\mathbb{S}$ matrix is unitary and symmetric. To show this, we need the definition of the *Wronskian* relation between two functions of one variable:

$$\mathcal{W}(a, b) = ab' - a'b, \tag{2.46}$$

where the prime means differentiation. The Wronskian between two matrix-functions of one variable is defined as

$$\mathcal{W}(\mathbb{A}, \mathbb{B}) = \mathbb{A}^T\mathbb{B}' - \mathbb{A}'^T\mathbb{B} \tag{2.47}$$

Important properties of Wronskian relations are antisymmetricity and bilinearity:

$$\mathcal{W}(\mathbb{A}, \mathbb{B}) = -\mathcal{W}(\mathbb{B}, \mathbb{A})^T \tag{2.48}$$

$$\mathcal{W}(\mathbb{A}\mathbb{C}, \mathbb{B}\mathbb{D}) = \mathbb{C}^T\mathcal{W}(\mathbb{A}, \mathbb{B})\mathbb{D} \tag{2.49}$$

where $\mathbb{C}$ and $\mathbb{D}$ are arbitrary, possibly complex, constant matrices.

Using these definitions and properties, it is found that

$$\mathcal{W}(\mathbb{F}, \mathbb{F}) = \mathcal{W}(\mathbb{G}, \mathbb{G}) = \mathbf{0} \tag{2.50}$$

$$\mathcal{W}(\mathbb{F}, \mathbb{G}) = \mathbf{1} \tag{2.51}$$

From $\mathcal{W}(\Psi, \Psi) = \mathbf{0}$ and the bilinearity of the Wronskian follows

$$\mathbb{S}^T\mathcal{W}(\mathbb{F}, \mathbb{G})^T - \mathcal{W}(\mathbb{F}, \mathbb{G})\mathbb{S} = 0, \tag{2.52}$$

and thus that $\mathbb{S}^T = \mathbb{S}$. If the potential is assumed to be real and symmetric, it can be shown that both Ψ and Ψ^* are solutions of the SE. Then, one can use Green's symmetrical theorem[9] to show that $\mathcal{W}(\Psi, \Psi^*) = \mathbf{0}$. From this relation, it is found that $\mathbb{S}^{-1} = \mathbb{S}^T$. This proves the symmetricity and unitarity of the $\mathbb{S}$ matrix.

2.6 Analytic solution for a constant potential

In this section, a (hypothetic) two-channel problem, with a constant potential is solved. The solutions given are the regular solutions. The irregular ones can be calculated in the same way, imposing the appropriate boundary or initial conditions.

2.6.1 A diagonal potential

We want to solve the CC equation for a two channel problem, with a diagonal $\mathbb{W}$ matrix:

$$\begin{bmatrix} u''_{11}(R) & u''_{12}(R) \\ u''_{21}(R) & u''_{22}(R) \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{11}(R) & u_{12}(R) \\ u_{21}(R) & u_{22}(R) \end{bmatrix}. \tag{2.53}$$

The two solutions of the SE are then given by [see Eq. (2.11)]

$$\Psi_j(R, \vec{r}) = u_{1j}(R)\phi_1(\vec{r}) + u_{2j}(R)\phi_2(\vec{r}), \text{with } j = 1, 2 \tag{2.54}$$

The general solutions of these differential equations are

$$\begin{aligned} u_{1j}(R) &= A\sin(R) + B\cos(R) \\ u_{2j}(R) &= C\exp(R) + D\exp(-R). \end{aligned} \tag{2.55}$$

The coefficients $A, B, C,$ and D follow from the imposed boundary or initial conditions. If we impose the general initial conditions

$$\begin{aligned} \mathbb{U}(0) &= \begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \\ \mathbb{U}'(0) &= \begin{bmatrix} c_1 & c_2 \\ d_1 & d_2 \end{bmatrix}, \end{aligned} \tag{2.56}$$

we find the general solution

$$u_{1j}(R) = c_j \sin(R) + a_j \cos(R) \tag{2.57}$$

$$u_{2j}(R) = \frac{1}{2}(b_j + d_j)\exp(R) + \frac{1}{2}(b_j - d_j)\exp(-R) \tag{2.58}$$

The initial conditions for the regular set of solutions are

$$\begin{aligned} \mathbb{U}(0) &= \mathbf{0} \\ \mathbb{U}'(0) &= \mathbf{1}, \end{aligned} \tag{2.59}$$

which gives

$$\begin{aligned} u_{11}(R) &= \sin(R) \\ u_{21}(R) &= 0 \\ u_{12}(R) &= 0 \\ u_{22}(R) &= \frac{1}{2}[\exp(R) - \exp(-R)]. \end{aligned} \tag{2.60}$$

The solutions to the SE are then

$$\begin{aligned} \Psi_1(R, \vec{r}) &= \sin(R)\phi_1(\vec{r}) \\ \Psi_2(R, \vec{r}) &= \frac{1}{2}[\exp(R) - \exp(-R)]\phi_2(\vec{r}). \end{aligned} \tag{2.61}$$

So in the case of a diagonal potential, there is no mixing of channel eigenfunctions.

2.6.2 A non-diagonal potential

Now, suppose that the $\mathbb{W}$ matrix in Eq. (2.53) is given by

$$\tilde{\mathbb{W}} = \begin{bmatrix} -\cos(\alpha) & \sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \tag{2.62}$$

This matrix can be diagonalized: $\tilde{\mathbb{W}} = \mathbb{T}\mathbb{W}\mathbb{T}^T$, where $\mathbb{T}$ is the orthogonal matrix:

$$\mathbb{T} = \begin{pmatrix} a & b \\ b & -a \end{pmatrix}, \tag{2.63}$$

with

$$\begin{aligned} a &= \frac{\sin(\alpha)}{\sqrt{2[1+\cos(\alpha)]}} \\ b &= \frac{1+\cos(\alpha)}{\sqrt{2[1+\cos(\alpha)]}}. \end{aligned}$$

The equation $\mathbb{U}''(R) = \mathbb{W}\mathbb{U}(R)$ has already been solved. We can multiply this equation from the left by $\mathbb{T}$, and insert $\mathbb{T}^T\mathbb{T} = \mathbf{1}$. Then we get

$$\mathbb{T}\mathbb{U}''(R) = \mathbb{T}\mathbb{W}\mathbb{T}^T\mathbb{T}\mathbb{U}(R) = \tilde{\mathbb{W}}\mathbb{T}\mathbb{U}(R) \tag{2.64}$$

If we now define

$$\begin{aligned} \tilde{\mathbb{U}}''(R) &= \mathbb{T}\mathbb{U}''(R) \\ \tilde{\mathbb{U}}'(R) &= \mathbb{T}\mathbb{U}'(R) \\ \tilde{\mathbb{U}}(R) &= \mathbb{T}\mathbb{U}(R), \end{aligned} \tag{2.65}$$

we see, that $\tilde{\mathbb{U}}(R)$ is the solution to the CC equation with the non-diagonal potential. The general form of $\tilde{\mathbb{U}}(R)$ can be found from Eqs. (2.57) and (2.63). After that, we can again impose the initial conditions for the regular set of solutions [Eq. (2.59)]. Then we find:

$$\begin{aligned}
\tilde{u}_{11}(R) &= a^2 \sin(R) + \frac{b^2}{2}[\exp(R) - \exp(-R)] \\
\tilde{u}_{21}(R) &= ab \sin(R) - \frac{ab}{2}[\exp(R) - \exp(-R)] \\
\tilde{u}_{12}(R) &= ab \sin(R) - \frac{ab}{2}[\exp(R) - \exp(-R)] \\
\tilde{u}_{22}(R) &= b^2 \sin(R) + \frac{a^2}{2}[\exp(R) - \exp(-R)].
\end{aligned} \tag{2.66}$$

In the case of a non-diagonal potential, the solutions of the SE become linear combinations of the different channel eigenfunctions. When R becomes large, the factor $\exp(R)$ becomes dominant in all terms of the solution matrix $\tilde{\mathbb{U}}$, so that asymptotically the solutions of the SE become

$$\begin{aligned}
\Psi_1(R,\vec{r}) &\overset{R\to\infty}{\sim} \frac{b}{2}\exp(R)[b\phi_1(\vec{r}) - a\phi_2(\vec{r})] \\
\Psi_2(R,\vec{r}) &\overset{R\to\infty}{\sim} -\frac{a}{2}\exp(R)[b\phi_1(\vec{r}) - a\phi_2(\vec{r})],
\end{aligned} \tag{2.67}$$

i.e. they grow exponentially and become linearly independent.

If we now impose the boundary conditions [Eq. (2.43)], instead of the initial conditions [Eq. (2.59)], we find the solutions:

$$\begin{aligned}
\tilde{u}_{11}(R) &= a^2 f_{\text{osc}}(R) + b^2 f_{\text{exp}}(R) \\
\tilde{u}_{21}(R) &= ab[f_{\text{osc}}(R) - f_{\text{exp}}(R)] \\
\tilde{u}_{12}(R) &= ab[f_{\text{osc}}(R) - f_{\text{exp}}(R)] \\
\tilde{u}_{22}(R) &= b^2 f_{\text{osc}}(R) + a^2 f_{\text{exp}}(R),
\end{aligned} \tag{2.68}$$

where

$$\begin{aligned}
f_{\text{osc}}(R) &= \frac{\sin(R-A)}{\sin(B-A)} \\
f_{\text{exp}}(R) &= \frac{\sinh(R-A)}{\sinh(B-A)}.
\end{aligned}$$

The functions f_{osc} and f_{exp} are plotted in figure 2.2, for $A = 0$ and $B = 10$. The solutions are used only in the interval $[A, B]$, so the exponential function will never blow up. The amplitude of the oscillating function depends on the value $B - A$. Therefore, the two solutions [Eq. (2.54)] will always be linearly independent, regardless the value of B.

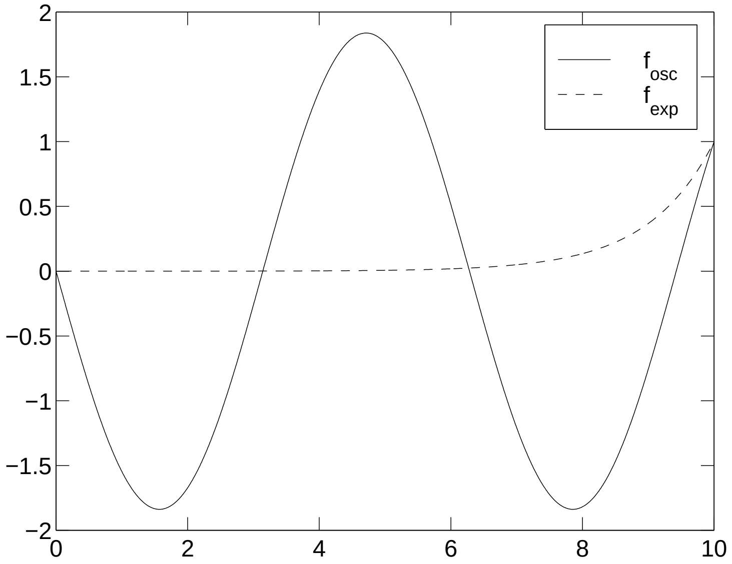

Figure 2.2: The functions f_{osc} and f_{exp} for $A = 0$ and $B = 10$.

From this it is seen that the CC equation has to be solved with the boundary conditions [Eqs. (2.43) and (2.44)]. In the literature, several methods are known to do this.[4–7] In the next chapter the algorithm we use is described. In chapters 4 and 5 optimizations of this algorithm are presented.

Chapter 3

The coupled channels algorithm

In this chapter, a description of the CC algorithm used by the DYNA program package, is given. It is used to construct DWs, the solutions to the Schrödinger equation in the external region [Fig. 1.2] of an atom-diatom reactive scattering problem, or of an inelastic atom-diatom scattering problem. There are several methods known to solve the coupled channels equation in inelastic scattering.[3]

One type of methods is called *adiabatic*. Adiabatic means in this context "slowly changing". With this, it is meant that the potential is relatively slowly changing with the scattering coordinate, and that it can be approximated by a piecewise constant potential. For a diagonal constant potential, the exact solution is known [Sec. 2.6]. Therefore, in this type of methods, the $\mathbb{W}$ matrix is evaluated and diagonalized at some R. This gives a basis of eigenfunctions that fit to the local potential, which is used in the coupled channels expansion in this step. Then the potential is approximated to be constant over a piece of the integration range. For this piece, the R-dependence of the wave function is then given by the exact solutions for this constant, diagonal potential. This procedure is then repeated for the rest of the integration range, resulting in a wave function over the total integration range.

The DYNA program package uses a *diabatic* method. Diabatic is the opposite of adiabatic, and means in this context "rapidly changing". This means that the potential is assumed to be (relatively) rapidly changing, and thus cannot be approximated to be constant or linear over a piece of the integration range. This results in a method, which solves the differential equations numerically, by evaluating the exact potential in some basis. This basis remains the same over the whole integration interval. The present method uses the rovibrational eigenfunctions of the diatom as basis functions.

As a new method a *quasi-adiabatic* type of method is presented [Chap. 5]. This method uses several different locally optimal bases, but not one basis at each step. The differential equations are solved numerically, there are no approximations. This makes it possible to integrate larger intervals in one basis than with a adiabatic method. Because of the use of a locally optimal basis, less basis functions are needed than with the diabatic type of method.

3.1 Propagating

The coupled channels equation has to be solved in each arrangement, for an interval of R, say $[A, B]$. The interaction region extends to $R = S$ [Fig. 1.2]. Because the CC equation is a second order differential equation, we can calculate two sets of solutions $\mathbb{U}_{\text{reg}}(R)$ and $\mathbb{U}_{\text{irreg}}$, the regular and irregular ones. A matrix relating the solution and its derivative in two points is called a propagator. For example, the log derivative (LD) propagator $\mathbb{Y}^{ba}$ is defined as:

$$\begin{aligned}\begin{bmatrix} -\mathbb{U}'(a) \\ \mathbb{U}'(b) \end{bmatrix} &= \qquad \mathbb{Y}^{ba} \qquad \begin{bmatrix} \mathbb{U}(a) \\ \mathbb{U}(b) \end{bmatrix} \\ &\equiv \begin{bmatrix} \mathbb{Y}_1^{ba} & \mathbb{Y}_2^{ba} \\ \mathbb{Y}_3^{ba} & \mathbb{Y}_4^{ba} \end{bmatrix} \begin{bmatrix} \mathbb{U}(a) \\ \mathbb{U}(b) \end{bmatrix}. \end{aligned} \tag{3.1}$$

This holds for both sets of solutions. When we define

$$\begin{aligned}\mathbb{U}^{ba} &= \begin{bmatrix} \mathbb{U}_{\text{irreg}}(a) & \mathbb{U}_{\text{reg}}(a) \\ \mathbb{U}_{\text{irreg}}(b) & \mathbb{U}_{\text{reg}}(b) \end{bmatrix} \\ \mathbb{U}'^{ba} &= \begin{bmatrix} -\mathbb{U}'_{\text{irreg}}(a) & -\mathbb{U}_{\text{reg}}(a) \\ \mathbb{U}'_{\text{irreg}}(b) & \mathbb{U}_{\text{reg}}(b) \end{bmatrix}\end{aligned} \tag{3.2}$$

we can write

$$\mathbb{Y}^{ba} = \mathbb{U}'^{ba} \mathbb{U}^{ba^{-1}}, \tag{3.3}$$

If these quantities were scalars, we could write $\mathbb{Y}^{ba} = \frac{d}{dR}(\ln \mathbb{U}^{ba})$ This explains the name log derivative propagator, or log derivative matrix. The boundary condition of the problem to be solved is given by

$$\mathbb{U}^{BA} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}, \tag{3.4}$$

so the LD propagator solves our boundary condition problem. When there are LD matrices for two adjacent intervals $[a, b]$, $\mathbb{Y}^{ba}$ and $[b, c]$, $\mathbb{Y}^{cb}$, these two can be combined to form one LD matrix $\mathbb{Y}^{ca}$ over the entire interval

$[a,c]$:

$$\begin{aligned} \mathbb{Y}_1^{ca} &= \mathbb{Y}_1^{ba} - \mathbb{Y}_2^{ba}[\mathbb{Y}_4^{ba} + \mathbb{Y}_1^{cb}]^{-1}\mathbb{Y}_3^{ba} \\ \mathbb{Y}_2^{ca} &= -\mathbb{Y}_2^{ba}[\mathbb{Y}_4^{ba} + \mathbb{Y}_1^{cb}]^{-1}\mathbb{Y}_2^{cb} \\ \mathbb{Y}_3^{ca} &= -\mathbb{Y}_3^{cb}[\mathbb{Y}_4^{ba} + \mathbb{Y}_1^{cb}]^{-1}\mathbb{Y}_3^{ba} \\ \mathbb{Y}_4^{ca} &= \mathbb{Y}_4^{cb} - \mathbb{Y}_3^{cb}[\mathbb{Y}_4^{ba} + \mathbb{Y}_1^{cb}]^{-1}\mathbb{Y}_2^{cb}. \end{aligned} \tag{3.5}$$

Because we use a routine to solve initial value problems, rather than boundary condition problems, we have to define an other type of propagator, the Cauchy propagator:

$$\begin{bmatrix} \mathbb{U}(b) \\ \mathbb{U}'(b) \end{bmatrix} = \begin{bmatrix} \mathbb{C}_1^{ba} & \mathbb{C}_2^{ba} \\ \mathbb{C}_3^{ba} & \mathbb{C}_4^{ba} \end{bmatrix} \begin{bmatrix} \mathbb{U}(a) \\ \mathbb{U}'(a) \end{bmatrix}. \tag{3.7}$$

This holds again for both sets of solutions. It is easy to see that two propagators over adjacent intervals $[a,b]$ and $[b,c]$ can be combined to one propagator:

$$\mathbb{C}^{ca} = \mathbb{C}^{cb}\mathbb{C}^{ba}. \tag{3.8}$$

This propagator solves the CC equation with the initial conditions

$$\begin{bmatrix} \mathbb{U}_1(A) & \mathbb{U}_2(A) \\ \mathbb{U}_1'(A) & \mathbb{U}_2'(A) \end{bmatrix} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}, \tag{3.9}$$

where $\mathbb{U}_1$ and $\mathbb{U}_2$ are the two sets of solutions. The relation between the Cauchy and LD propagator is given by (omitting the interval superscripts):

$$\begin{aligned} \mathbb{Y}_1 &= \mathbb{C}_2^{-1}\mathbb{C}_1 \\ \mathbb{Y}_2 &= -\mathbb{C}_2^{-1} \\ \mathbb{Y}_3 &= \mathbb{C}_3 - \mathbb{C}_4\mathbb{C}_2^{-1}\mathbb{C}_1 \\ \mathbb{Y}_4 &= \mathbb{C}_4\mathbb{C}_2^{-1}. \end{aligned} \tag{3.10}$$

The DYNA program package uses a Runge-Kutta-Nystrom method, implemented in the NAG Fortran Library[2] to solve this initial value problem over a short interval. This results in the Cauchy propagator over this interval, which is transformed into a LD propagator over the interval. This procedure is repeated for the rest of the total propagation interval (the grey parts in fig. 1.2), thus dividing the propagation interval in steps. When a Cauchy propagator over a step is calculated, it is transformed into the LD propagator over the step. This LD propagator is then combined with the LD propagator over the previous interval into one propagator over the total propagated interval.

3.2 Overlap between outer and interaction region

In the overlapping part of the interaction region and the outer region, some special care has to be taken. The DWs are used as basis functions in the variational calculation of the reactive $\mathbb{S}$ matrix. Therefore the DWs must be evaluated at a grid in this overlapping part.

So not only the log derivative matrix over the interval $[A, S]$ is needed. Therefore, the LD matrices are calculated over the intervals as defined by this grid. Then these intermediate LD matrices are stored to disk. This makes it possible to evaluate the DWs.

3.3 Calculation of the inelastic $\mathbb{S}$ matrix

From the regular DWs it is possible to calculate an inelastic $\mathbb{S}$ matrix. If the DWs are calculated to be used as basis functions in a reactive calculations, this "artificial" $\mathbb{S}$ matrix is without physical meaning.

From the boundary condition $\mathbb{U}(A) = \mathbf{0}$ ($A = 0$ for true inelastic scattering) it is found that

$$\mathbb{Y}_4^{BA} = \mathbb{U}'(B)\mathbb{U}^{-1}(B). \tag{3.11}$$

From the matching condition [Eq. (2.42)] one can deduce that

$$\mathbb{S} = [-\mathbb{U}'_{\text{out}} + \mathbb{Y}_4^{BA}\mathbb{U}_{\text{out}}]^{-1} \times [-\mathbb{U}'_{\text{in}} + \mathbb{Y}_4^{BA}\mathbb{U}_{\text{in}}], \tag{3.12}$$

where the sets of asymptotic solutions $\mathbb{U}_{\text{in}}$ and $\mathbb{U}_{\text{out}}$ must be evaluated at the matching point B.

3.4 The total reactive $\mathbb{S}$ matrix

The total reactive $\mathbb{S}$ matrix is the result of the variational calculation, in which the DWs are used as basis functions. A detailed description of the variational method can be found in the literature.[9,10]

3.5 Implementation of the coupled channels algorithm in DYNA

The coupled channels algorithm in DYNA solves the CC equation on the interval $[A, B]$, where the interaction region extends to S, between A and B (see also fig. 1.2). There are four steps.

1. Calculation of the log derivative matrix over the interval $[A, S]$, and storage of the intermediate results on a grid in this interval. These intermediate results are necessary to evaluate the DWs on a grid in the interaction region.

2. Calculation of the LD matrix over the interval $[S, B]$, and storage of the LD matrices over this interval.

3. Combination of the LD matrices over the two previous intervals into one matrix over the entire interval $[A, B]$.

4. Calculation of the inelastic $\mathbb{S}$ matrix. This is done for test purposes only, the quality of this artificial $\mathbb{S}$ matrix is a measure for the quality of the DWs. This has been used in the testing of the optimization of the algorithm, presented in chapter 4 and 5.

The first step is done by the routine JAB, which calculates the grid on which the wave function has to be calculated. This routine calls the routine INTERVAL which returns the LD matrices over the entire interval $[A, S]$. The routine JAB stores these results to disk.

The routine INTERVAL calculates the LD propagators over each interval in the calculated grid. It stores the intermediate LD matrices to disk, to make it possible to recalculate the DWs on this grid.

The LD matrices over a given interval, say $[a, b]$, are calculated with the routine SECPROP. This routine is called by INTERVAL in "step-mode", i.e. INTERVAL uses SECPROP to calculate the LD matrix over a small interval and to combine this result with the results of the previous intervals. It works in the following way.

It first calls the routine GETSTEP to calculate the size of the interval, over which a Cauchy propagator can safely be calculated. It then calls the routine CAUCHY, which calculates the Cauchy matrices over this small interval, and transforms this matrix into a LD matrix over the same interval. If this interval is smaller than the propagation interval $[a, b]$, it repeats this procedure, and combines the LD matrices over the different intervals into one LD matrix over the total interval $[a, b]$.

The routine GETSTEP needs (an estimate of) the largest eigenvalue of the $\mathbb{W}$ matrix at the begin point of the interval, and a tolerance. This tolerance is a measure for the maximum allowed relative error in the solutions. The solutions will contain exponentially growing terms. If the square root of the largest eigenvalue is given by k, the fastest growing solution will be of the order of $\exp(k * R)$. The maximum value the solution may reach, is defined to be

$$\text{maxval} = 10^{-4}\text{tolerance}/\text{machine precision} \tag{3.13}$$

This guarantees that the number of significant figures in the solution is at least four more than required by the setting of the tolerance. This default has been chosen to be sure that the required accuracy in the $\mathbb{S}$ matrix can be made. The length of the interval to be propagated is then calculated by

$$h = \ln(\text{maxval})/k \tag{3.14}$$

The routine that performs the actual propagation is CAUCHY. Using the boundary conditions at the beginning of the interval [Eq. (3.9)], it calculates the solutions at the end of the propagation interval. To solve the set of coupled differential equations, it uses the NAG routine D02LAF. This routine uses a Runge Kutta Nystrom method to solve the system of equations. From the definition of the Cauchy propagator [Eq. (3.7)] and the initial conditions [Eq. (3.9)], it is easily seen that the Cauchy matrices are given by:

$$\begin{bmatrix} \mathbb{U}_1(b) & \mathbb{U}_2(b) \\ \mathbb{U}_1'(b) & \mathbb{U}_2'(b) \end{bmatrix} = \begin{bmatrix} \mathbb{C}_1 & \mathbb{C}_2 \\ \mathbb{C}_3 & \mathbb{C}_4 \end{bmatrix} \tag{3.15}$$

Computationally, the second step (on the interval $[S, B]$) is almost equal to the first step (on the interval $[A, S]$). The difference is that there is no need to store intermediate results. The only thing needed is the LD matrix over the total interval $[S, B]$. This is computed by the routine JSECTOR which calls the routine SECPROP and stores the results to disk.

The third and fourth step, the combination of the two intervals $[A, S]$ and $[S, B]$ and the calculation of the $\mathbb{S}$ matrix is done by one routine, JDIST2. This routine reads the LD matrices from disk, combines them, and matches them to the asymptotic solutions. It calculates the reactance matrix $\mathbb{K}$ and the matrices $\mathbb{Q}_1$ and $\mathbb{Q}_2$, defined in Ref. 9, from which the artificial inelastic scattering matrix $\mathbb{S}$ can be calculated: $\mathbb{S} = (\mathbb{Q}_1 + i\mathbb{Q}_2)(\mathbb{Q}_1 - i\mathbb{Q}_2)^T$. The routine JDIST2 stores the $\mathbb{Q}$ matrices to disk. The reactance matrix $\mathbb{K}$ is a matrix that is related to the $\mathbb{S}$ matrix. The relation between them is given by $\mathbb{S} = (\mathbf{1} + i\mathbb{K})(\mathbf{1} - i\mathbb{K})^{-1}$ The advantage of the reactance matrix is that is is real.

The program package has a flexible design. It consists of a number of MATLAB routines and FORTRAN programs that take parameters. The user can solve different types of atom-diatom scattering problems (collinear or $3D, J = 0$, inelastic or reactive, with or without symmetry) by choosing the appropriate parameters for the programs, and choosing the MATLAB routines, needed to solve the problem. The user writes "jobs", i.e. shell scripts, which define what programs to run with what parameters.

All the routines described above are in MATLAB, and must be called from a MATLAB job (a shell script containing a MATLAB program). The NAG routine mentioned above is in FORTRAN, and also the routines to evaluate the potential are in FORTRAN. This part is also flexible, the user can supply the name of the potential routine to use.

Chapter 4

Optimization of the use of D02LAF

The present method uses a NAG[2] routine (D02LAF) to solve an initial value problem over a short interval, and the new method presented in this report will use this numerical routine also. The evaluation of the $\mathbb{W}$ matrix is computationally very expensive. Therefore an optimal parameter setting is such that the routine D02LAF uses as few $\mathbb{W}$ matrix evaluations as possible.

4.1 Description of the NAG routine D02LAF

The routine D02LAF is a routine to solve a non-stiff system of second order ordinary differential equations. It uses two Runge-Kutta-Nystrom formula-pairs, one of high (12th and 10th order) and one of lower (6th and 4th) order. The user can, by means of a parameter to the setup routine described later, state which pair he or she wants to use. Also requirements on the accuracy of the solution (error control) and the mode of operation (stepwise or interval mode) are selected by the call to this setup routine

More specifically, the routine integrates a system of equations of the form

$$y_i''(x) = f_i(x, y_1, y_2, \ldots, y_n) \qquad i = 1, 2, \ldots, n \tag{4.1}$$

from $x = x_{\text{begin}}$ to $x = x_{\text{end}}$, where n is the number of equations in the system.

The user must supply the initial conditions $y_1, y_2, \ldots, y_n, y_1', y_2', \ldots, y_n'$ at $x = x_{\text{begin}}$ and a subroutine that evaluates the system-defining functions f_i, given the point x and the values $y_1, y_2, \ldots, y_n$

The precise specification of this routine, and its setup routine D02LXF is given in appendix A. The most important parameters of D02LAF are a user-supplied function to evaluate the second derivatives f_i, the interval that must be integrated, and the initial conditions y_i and y_i'. The accuracy the user wants in the answer, can be set by a call to the setup routine. This routine has 4 parameters to control the accuracy of D02LAF:

- A hint for the next step size to be attempted by D02LAF (`H`).
- The tolerance (`TOL`).
- Threshold values on the solution (`THRES`).
- Threshold values on the first derivatives of the solution (`THRESP`).

The last three parameters, together with the local error estimates of D02LAF, determine whether an attempted step will be accepted or not. This works in the following manner. At each step, D02LAF makes an estimate of the local errors made in that step. We designate the local error estimates on the solution by $E_1(i)$ and the error estimates on the derivatives by $E_2(i)$. The step will be accepted if the next conditions are satisfied:

$$\max_{1\leq i\leq n}\left(\frac{E_1(i)}{\max(\mathtt{THRES}(i),|y_i|)}\right)\leq \mathtt{TOL} \tag{4.2}$$

$$\max_{1\leq i\leq n}\left(\frac{E_2(i)}{\max(\mathtt{THRESP}(i),|y_i'|)}\right)\leq \mathtt{TOL} \tag{4.3}$$

This means that `TOL` is a bound for the minimum of the relative and the absolute local error estimate in each step. The routine controls only the local error, the overall error over more than one step can accumulate in several system dependent ways. The code is so designed that a reduction in `TOL` should lead to an approximately proportional reduction in the error in the solution, but this is not guaranteed.

The routine cannot guarantee a bound on either the relative or the absolute local error. The relative error can become very large for solutions that are very near to zero, and for rapidly growing solutions, the absolute error can become very large. Therefore, the routine uses a mixed kind of error control. By means of the parameters `THRES` and `THRESP` the user can define for what values of the solution and its derivative a relative or absolute error test should be used.

In the optimization of the use of this routines, it is important to control the error and so the parameters of most importance in the study will be the error controlling parameters `TOL`, `THRES`, and `THRESP` of the initialization routine D02LXF. Furthermore, giving a good hint on the step size to attempt can save a lot of unsuccessful steps, and thus save a lot of evaluations of the functions f_i. In our problem, these functions are the evaluation of $\mathbb{WU}$, an expensive evaluation of $\mathbb{W}$, and a matrix multiplication. The reduction of the number of function evaluations is thus a good point to optimize.

4.2 How the routine was used before the optimization

Before the optimization, the routine CAUCHY called the NAG routines with more or less standard parameter settings. These parameter settings might not be optimal.

The routine CAUCHY used D02LAF to solve all the different initial conditions at the same time, by putting all the columns of the solution matrix $\mathbb{U}$ and its derivative $\mathbb{U}'$ into two long column vectors, of length $2n^2$ (there are $2n$ different initial conditions, and n equations per initial condition).

As a hint for the step size, the length of the total interval was used. One expects that this is a rather unfortunate setting, because one does not expect that this interval can be propagated in one step.

The thresholds were all set to unity. This also is not very efficient. We know that our solutions are wave-like, of the form $\exp(ikx)$, with wavelength $2\pi/k$. So, we know that the derivatives are a factor k larger than the solutions. It might therefore be reasonable to use a guess of the wavelength of the solutions in the thresholds for the derivatives. The choice of ones in the thresholds on the solutions is good, since it means using a absolute error test on the oscillating solutions in the open channels, and a relative error test on the exponential growing solutions in the closed channels.

4.3 Tests

To test the effect of the different parameters on the results, and the time efficiency of D02LAF, a simple test equation has been used. Because the solutions to our problem are basically sine, cosine, and exponential functions, we used the one dimensional test equation $y'' = wy$, with w positive for exponential solutions and negative for sine and cosine solutions. The wavelength of the oscillating solutions is equal to $2\pi/\sqrt{|w|}$.

From the first tests, it was seen that the routine uses an approximately equal step size for both the sine/cosine and the exponential problem (with the same absolute value $|w|$). If the boundary conditions were set so that the exact solution was a decaying solution:

$$y''(x) = k^2 y(x), \text{ with } y(0) = 1 \text{ and } y'(0) = -k, \tag{4.4}$$

it was found that the solution found was a linear combination of the decaying and a very small part of the growing exponential:

$$\begin{aligned} y_{\text{exact}}(x) &= \exp(-kx) \\ y_{\text{computed}}(x) &= y_{\text{exact}} + \text{error} \\ \text{error} &\approx \alpha \exp(kx) \end{aligned} \tag{4.5}$$

When the tolerance was set to 10^{-6}, α was typically of the order of 10^{-11}. The parameter α did not increase over the interval when the integration interval became larger, up to 10 wavelengths. Although the growing part is very small, it becomes dominant after about 2 wavelengths

We have seen that the closed channels (with positive values of k) are the trouble causing part in our problem. These closed channels correspond to exponential solutions, which cause numerical problems. Therefore we decided to use the growing exponential test equation.

The test calculations carried out were the following. In each calculation the test system was integrated over the interval $[0, 2\pi]$. For each k in the set $\{4, 6, 8, 10\}$ a set of calculations with different tolerances was done. The tolerances used were 10^{-p}, $p = 1, 2, \ldots, 14$. The thresholds were all set to 1, and no hint for the step size was given. It is possible not to give such a hint, but giving a good hint may speed up the integration.

From the results some formulae could be deduced. Given the length of the integration interval, the wavelength of the problem and the desired bound on the error at the end of the interval, there are formulae to calculate a hint for the step size to use, and the tolerance one should use.

These formulae were tested using the DYNA program package on the linear collision of H + H_2 described in chapter 2 and by Secrest and Johnson.[8,11] Results for the inelastic transition probabilities of this system are also found in Ref. 11. We did a calculation with tolerance 10^{-6}, with all the old parameter settings, and three test calculations with the new formulae, with tolerances 10^{-1}, 10^{-4}, and 10^{-6}.

4.4 Results

In figure 4.1, the relationship between the tolerance and the relative error at the end of the propagation interval is plotted.

From this figure is seen, that the relationship between the logarithms of the tolerance and the error are linear over more than 8 orders of magnitude for the tolerance. There appears to be a maximum attainable accuracy of approximately $10^{-14} \approx 100\times$ machine precision.

It is also seen that the error grows slightly, when the wavelength in the problem gets smaller, i.e. when k increases. The explanation for this is, that the integration range is the same in all tests, namely $[0, 2\pi]$. So for $k = 10$, this interval is 10 wavelengths long, and for $k = 4$ only 4. There is also about a factor of 2.5 between the errors of $k = 4$ and $k = 10$.

Because our problem contains both sine/cosine and exponential solutions, in practice, we can not propagate over intervals longer then a few wavelengths. The solutions would become linearly dependent, as explained in the previous chapter. For our formulae, this slight dependence of the error on the ratio of wavelength and interval-length is not so important, and

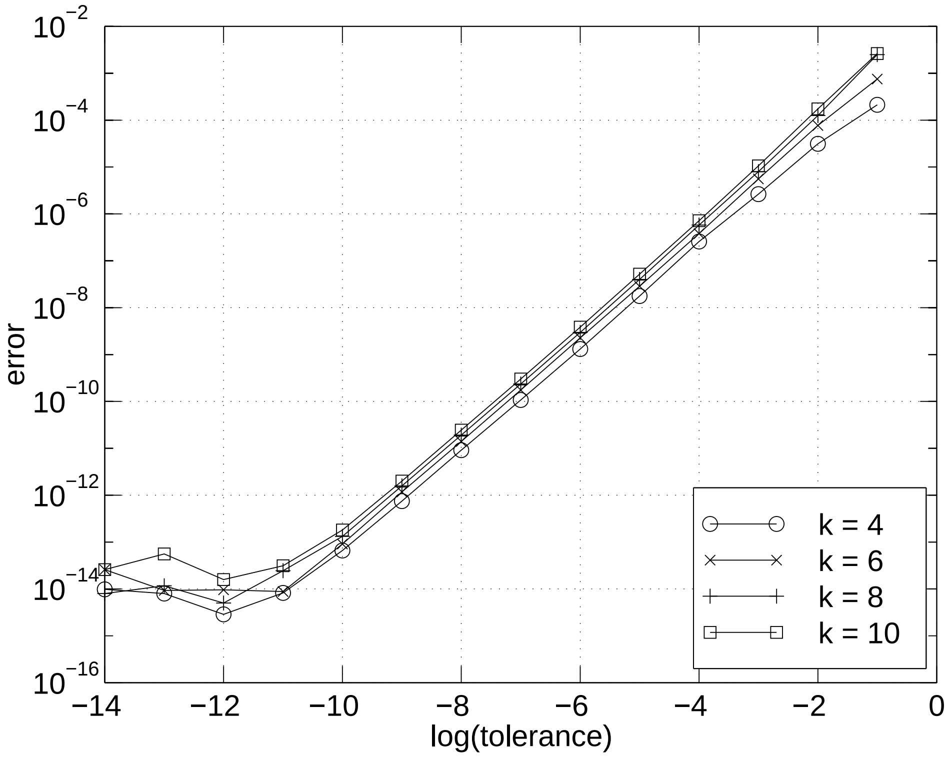

Figure 4.1: Relative error at the end of the integration interval, as a function of tolerance, for different values of the wavelength ($= 2\pi/k$).

can be neglected.

We can fit lines through the data points in figure 4.1. In table 4.1, the coefficients a and b are given for the four lines $\log(\text{error}) = a\log(\text{tolerance}) + b$. These are fits through the data points for tolerances between 10^{-2} and 10^{-10}, the straight part of the lines. Also the mean value is given.

In figure 4.2, the results for the step size D02LAF takes (h) are displayed as a function of the tolerance. The step sizes are displayed as dimensionless quantities c, where c is defined as $c = hk$. The step sizes displayed are the ones that the routine takes after the k value of the problem has been estimated. This takes a few steps.

We expect that this measure makes the step size independent of the wavelength, because the routine will always need a certain number of steps per wavelength. For the routine, it does not matter in what units you state your problem. It makes no difference whether you want to solve a problem with integration interval $[0, a]$ and wavelength $a/2$ or a problem with interval $[0, 2a]$ and wavelength a. These two problems are numerically equivalent.

Table 4.1: Least squares fits to the lines in fig. 4.1.

k	a	b
4	1.0791	-2.3564
6	1.1066	-1.9679
8	1.0953	-1.8602
10	1.0967	-1.7379
mean	1.0944	-1.9806

In figure 4.2 it is seen that the stepsize depends directly on the tolerance (τ), the log-log relationship is linear for over 9 orders of magnitude. Furthermore, we see that the stepsize is directly proportional to the inverse of the wavelength, as was expected.

The least squares fit to the data in figure 4.2 gives as relationship between c and the tolerance: $\log(c) = 0.1008 \log(\text{tolerance}) + 0.9737$.

For most efficient operation, D02LAF requires a hint for the step size. As hint we use the stepsize that the NAG takes after the first few steps in the test problem. In these first few steps, the routine estimates the wavelength of the problem it is solving.

We had to choose something for the b parameter in the relation between the tolerance and the error. The value of b depends on the ratio of the wavelength and the length of the interval. In our problem, we can not propagate more than a few wavelengths, so we chose 2. This agrees with about 5 or 6 wavelengths per interval.

If one wants the result with an maximum relative error ϵ at the end of the interval, one has to choose

$$\tau = (100\ \epsilon)^{0.909} \tag{4.6}$$

and the hint for the step size has to be

$$h = 9.41239\ (\tau^{0.1008})/k \tag{4.7}$$

In our multi-channel problems, we use an estimate of the square root of the largest absolute eigenvalue of the $\mathbb{W}$ matrix for k.

4.5 Application to the test system H + H_2

The results from the preceding paragraph were tested on the collinear inelastic collision of H and H_2, as described by Stechel, Secrest, and Johnson.[7,8] The integration interval was $[20, 100]$, and the scattering energy 3.0, the total energy was 6.0, in the units determined by the scaling from Secrest and

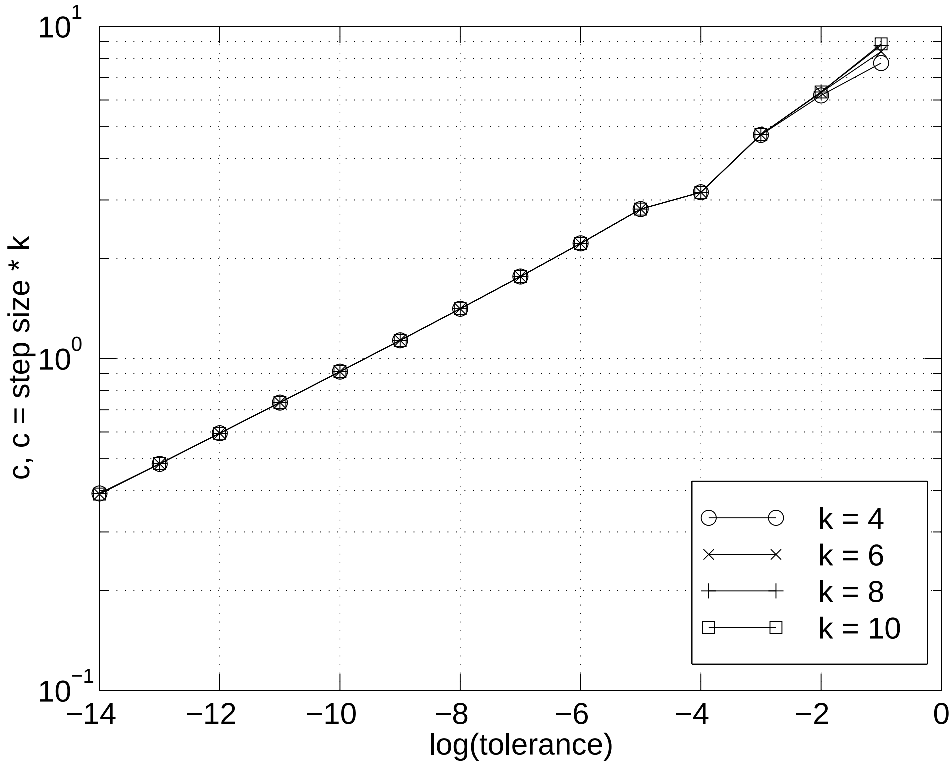

Figure 4.2: Dimensionless step size as function of the tolerance.

Johnson. These energies result in a calculation with 6 channels, 3 asymptotically open, and 3 closed channels. Since the interval is too large to propagate directly with the NAG routine, we have to use the propagators as described in chapter 3.

Five calculations were performed. The first calculation was with all the old parameter settings, as described in section 4.2. In this calculation the tolerance was set to 10^{-6}. The calculations, in which the new parameter settings were used, were with three different values for the expected error at the end of the interval (ϵ), 10^{-1}, 10^{-4}, and 10^{-6}. Finally, one reference calculation was performed. This calculation used the new parameter setting, and an ϵ-value of 10^{-8}. The errors in the $\mathbb{S}$ matrix, reported in table 4.2, were calculated with respect to the result of this calculation.

Secrest and Jonhson[8] report transition probabilities for this system with high precision, eight significant figures. These results are reproduced in all figures by our old calculation, and by our new calculations with $\epsilon = 10^{-6}$ and 10^{-8}. In the literature, also $\mathbb{S}$ matrix elements are reported.[7] These only have four significant figures. Our calculations with $\epsilon = 10^{-4}$, 10^{-6}, and

Table 4.2: Results for the test of the NAG optimization formulae.

Calculation	Error	Number $\mathbb{W}$ eval.	propagation time (sec)
old	4.26×10^{-8}	2579	3.33
new, $\epsilon = 10^{-1}$	3.62×10^{-1}	679	1.27
new, $\epsilon = 10^{-4}$	5.08×10^{-5}	1164	1.78
new, $\epsilon = 10^{-6}$	2.93×10^{-7}	1706	2.44

10^{-8} reproduce these values.

It is seen that reducing ϵ with one order of magnitude gives an error approximately one order larger. The errors in the transition probabilities are much smaller then expected, but the errors in the $\mathbb{S}$ matrix are as expected. This can be explained by the fact that transition probabilities are usually better then $\mathbb{S}$ matrix elements. This is also found in the literature.[7]

The new method requires about 33% fewer potential evaluations to get the same accuracy. The reduction in user time is not equally large, about 27%. This may be explained by the fact that these tests are relatively small, there is a lot of overhead. The I/O that has to be done during the propagation, is independent of the number of channels, and thus takes a relatively large amount of time in small tests. But from the reduction of the number of potential evaluations, it is seen that the hint for the step size is good. From a careful analysis of the number of potential evaluations that the NAG routine needed, it was seen, that with the old method, it needed more then 30 evaluations in the first step of a Cauchy interval(i.e. the interval over which one Cauchy propagator is calculated), and 17 in each next step. With the new method, it needed 18 evaluations in the first step, and 17 in each next step. This means that the hint used in the old method was much too large, and that the first attempted step was always rejected. In the new method, this first step succeeds, which saves 17 potential evaluations per Cauchy interval.

Furthermore, the new method requires fewer steps. This can be explained by the better threshold settings for the derivatives. With the old settings, sometimes a relative error test was used for the open channels (oscillating solutions), which is a sharper error test than an absolute error test for these kind of solutions. Therefore the derivatives were calculated too accurately. This decreases the step size, and the routine took more steps than necessary.

Chapter 5

The quasi-adiabatic method

5.1 Introduction

In this chapter the second optimization to the coupled channels algorithm is presented. This optimization includes a new method to solve the CC equation on the interval $[S, B]$ [Fig. 1.2]. The first advantage of the new method is that it can be implemented as a black box routine. The number of parameters that has to be set by the user is minimized. The user can specify the desired accuracy of the resulting distorted waves. The second advantage is that it is faster than the method used before.

The new method solves the CC equation in the following way. The total integration interval $[S, B]$ is divided in steps. In each step a locally optimal basis is calculated (the quasi-adiabatic basis of the step). To calculate this quasi-adiabatic basis, first a set of primitive basis functions, to be used in this step, is determined for each step. This primitive basis consists of "primitive" basis functions, the eigenfunctions of the diatom [Eq. (2.29)]. The $\mathbb{W}$ matrix is then evaluated at the begin point of the step, in the primitive basis of the step. Diagonalization then gives the local eigenfunctions and thus the quasi-adiabatic basis of the step.

The Cauchy matrix is then calculated in the quasi-adiabatic basis, and transformed to the LD matrix over the step. This LD matrix is then transformed back to the primitive basis of the step. After LD matrices over all the steps have been calculated, they are combined in one LD matrix over the total propagation interval.

5.2 Theoretical description

5.2.1 Division of the interval in steps

The length of the steps and the number of functions to be included in the bases of the different steps are determined by 2 parameters, $E_{\mathrm{max,S}}$ and a tolerance parameter, τ. The tolerance parameter τ is intended to be a measure of the accuracy of the calculated DWs. The parameter $E_{\mathrm{max,S}}$ determines how many local channels will be included in the quasi-adiabatic basis of each step. A local channel will be included when its energy is less than $E_{\mathrm{max}}(R_i)$, where R_i is the begin point of i-th the step. It is assumed that E_{max} scales linearly with the scattering coordinate R:

$$E_{\mathrm{max}} = \frac{R-S}{B-S}E + \frac{B-R}{B-S}E_{\mathrm{max,S}} \tag{5.1}$$

where E is the total scattering energy and $E_{\mathrm{max,S}}$ is the value of $E_{\mathrm{max}}(R)$ at $R = S$. The length of a step (h_i) is determined in the same way as the length of a Cauchy step in chapter 3:

$$\begin{aligned} m_i &= 10^{-4} \times \tau/\epsilon \\ k_i &= \sqrt{2\mu[E_{\mathrm{max}}(R_i) - E]} \\ h_i &= \ln(m_i)/k_i, \end{aligned} \tag{5.2}$$

where m_i is the maximum value the solution may reach in the most strongly closed channel, ϵ is the machine precision, and k_i is an upper bound for the largest eigenvalue of the $\mathbb{W}$ matrix at $R = R_i$. The first step starts at S, i.e. $R_1 = S$ and the steps are adjacent: $R_{i+1} = R_i + h_i$. The tolerance for each step is now calculated from the total tolerance parameter τ and the interval length. If the i-th interval has length h_i, the tolerance to be reached on this step is

$$\tau_i = \tau\frac{h_i}{B-S}. \tag{5.3}$$

From this τ_i the tolerance parameter for the NAG routine is calculated.

5.2.2 Calculation of bases for each step

In this section, it is described how the bases in a certain step are calculated. Assume that the step is over the interval $[R_i, R_{i+1}]$. At the very beginning of the calculation a large set of primitive basis functions is calculated, using a program named CHANNEL. This basis will be referred to as the channel-basis. This set must be large enough to make sure that there enough primitive functions to describe all local channels in the first step accurately. In steps further outward, fewer local channels will be included and consequently also fewer primitive functions will be needed to describe them.

The quadrature used in the evaluation of the $\mathbb{W}$ matrix elements depends on the number of primitive basis functions. The grid in the r coordinate is equidistant, with a user definable number of points per shortest wavelength. In the coordinate θ a Gauss-Legendre quadrature is used, with $j_{\mathrm{max}} + 1 + n_{\mathrm{extra}}$ points, where j_{max} is the maximum value of the j quantum number in the set of functions, and n_{extra} defines how many extra points should be taken. When fewer primitive basis functions are used, the quadrature also requires fewer points. A reduction of the number of primitive basis functions with a factor a therefore means a reduction of an order of a^3 of the computing time needed to evaluate the $\mathbb{W}$ matrix. This is the reason why a different primitive basis is used in each step.

The primitive basis of the i-th step is calculated as follows. The $\mathbb{W}$ matrix is evaluated at R_i, in the channel-basis. This matrix is diagonalized:

$$\mathbb{W}^{\mathrm{channel}}\mathbb{T} = \mathbb{T}\Lambda \tag{5.4}$$

The matrix $\mathbb{T}$ is orthogonal since $\mathbb{W}^{\mathrm{channel}}$ is real and symmetric, and it defines the basis transformation between the channel basis and a basis of local eigenfunctions. The diagonal matrix Λ contains the eigenvalues of the $\mathbb{W}$ matrix, the energies of the local eigenfunctions. The columns of $\mathbb{T}$ give the coefficients of the primitive basis functions in the quasi-adiabatic basis. This quasi-adiabatic basis is normalized to unity, so for each column j of $\mathbb{T}$ holds

$$\sum_{i=1}^{N} \mathbb{T}_{ij}^2 = 1, \tag{5.5}$$

where N is the number of functions in the channel basis. For each local basis function j that has to be taken into account by the energy criterion $\Lambda_{jj} \leq E_{\max}(R_i)$, the primitive basis functions with the largest components in this local functions are put into the primitive basis set of the i-th step. This is done so that for each local function j holds that

$$\sum_{i} \mathbb{T}_{ij}^2 \geq \delta, \tag{5.6}$$

where i runs over all the functions in this new primitive set, and δ is the accuracy up to which the local channels are required to be represented. In the present method, δ has been set to 0.999.

The local basis of the step must now be recomputed using the smaller primitive basis of the step, to be able to use the smaller quadrature grid. This is efficient because of the large number of $\mathbb{W}$ matrix evaluations needed in the propagation (about 100 per step).

5.2.3 Propagation across one step

Evaluation of the $\mathbb{W}$ matrix at $R = R_i$ in the primitive basis of the i-th step, and diagonalization of this matrix yields the transformation matrix between the primitive and local basis of this step:

$$\mathbb{W}(R_i)\mathbb{T}_i = \mathbb{T}_i\Lambda_i \tag{5.7}$$

The $\mathbb{W}$ matrix in the local basis is given by

$$\tilde{\mathbb{W}}(R) = \mathbb{T}_i^T \mathbb{W}(R) \mathbb{T}_i. \tag{5.8}$$

We now want to solve the CC equation in the local basis:

$$\tilde{\mathbb{U}}''(R) = \tilde{\mathbb{W}}(R) \tilde{\mathbb{U}}(R). \tag{5.9}$$

The relation between the solutions of Eq. (5.9) and the solutions of the CC equation in the primitive basis [Eq. (2.34)] is given by:

$$\begin{aligned} \tilde{\mathbb{U}}(R) &= \mathbb{T}_i^T \mathbb{U}(R) \\ \tilde{\mathbb{U}}'(R) &= \mathbb{T}_i^T \mathbb{U}'(R) \\ \tilde{\mathbb{U}}''(R) &= \mathbb{T}_i^T \mathbb{U}''(R) \end{aligned} \tag{5.10}$$

Matrices with a tilde are in the quasi-adiabatic basis of the step, and matrices without the tilde are in the primitive basis of the i-th step. This gives for the log derivative matrices over this step:

$$\tilde{\mathbb{Y}}_k^{R_{i+1} R_i} = \mathbb{T}_i^T \mathbb{Y}_k^{R_{i+1} R_i} \mathbb{T}_i, \tag{5.11}$$

where $k = 1, 2, 3, 4$ for the different blocks of the LD propagator. Using the transformation matrix $\mathbb{T}_i$, it is possible to evaluate the local basis functions at the quadrature points. The primitive basis functions transform into the local ones in the same way as in Eq. (5.10). This makes it possible to evaluate the $\tilde{\mathbb{W}}$ matrix in the local basis, the quadrature used is not changed by the basis transformation. Now the Cauchy and LD propagator can be calculated in the quasi-adiabatic basis. The LD propagator is then transformed back to the primitive basis of the step. This back transformation is formally given by

$$\mathbb{Y}_k^{R_{i+1} R_i} = [\mathbb{T}_i^T]^{-1} \tilde{\mathbb{Y}}_k^{R_{i+1} R_i} \mathbb{T}^{-1}. \tag{5.12}$$

When $\mathbb{T}$ is square, there is no trouble, since $\mathbb{T}^T = \mathbb{T}^{-1}$. But when not all the local channels are used in the calculation, $\tilde{\mathbb{Y}}_k^{R_{i+1} R_i}$ is smaller than $\mathbb{Y}_k^{R_{i+1} R_i}$, and the part of the transformation matrix used is not square. Then the inverse does not exist formally. To be able to use the full, square transformation matrix, some approximation should be made for the parts of the LD matrices $\tilde{\mathbb{Y}}_k$ that are not calculated. From theory we know that for a constant diagonal $\mathbb{W}$ matrix the LD matrices are diagonal, with diagonal elements larger when the channel energy is larger. We assume that for the high energy channels not included the coupling with other channels is negligible, and that the potential is constant over the interval. Then we include the analytic solution for the diagonal constant $\mathbb{W}$ matrix for these high energy local channels in the LD matrices.

5.2.4 Combining the results of the different steps

In assembling the results of the different intervals, some special care has to be taken. It is impossible to use Eq. (3.5) directly, because of the different sets of primitive basis functions used. However, all primitive bases are subsets of the large channel basis.

In the matching of two sets of log derivative matrices over adjacent intervals, only the intersection of the two sets of basis functions is used. To match LD matrices over the intervals $[a, b]$ and $[b, c]$, use Eq. (3.5) with the following "reduced" matrices:

- $\bar{\mathbb{Y}}_1^{cb}$, $\bar{\mathbb{Y}}_4^{ba}$: take only the rows and columns of $\mathbb{Y}_1^{cb}$ and $\mathbb{Y}_4^{ba}$ respectively, corresponding with channels in the intersection.
- $\bar{\mathbb{Y}}_2^{ba}$, $\bar{\mathbb{Y}}_3^{cb}$: take only the columns of $\mathbb{Y}_2^{ba}$ and $\mathbb{Y}_3^{cb}$ respectively, corresponding with channels in the intersection.
- $\bar{\mathbb{Y}}_2^{cb}$, $\bar{\mathbb{Y}}_3^{ba}$: take only the rows of $\mathbb{Y}_2^{cb}$ and $\mathbb{Y}_3^{ba}$ respectively, corresponding with channels in the intersection.

The matrices $\mathbb{Y}_1^{ba}$ and $\mathbb{Y}_4^{cb}$ are used entirely.

The result is a LD matrix that connects the radial part of the wave function, expanded in a certain number of channels at S, with the radial part of the wave function at B, expanded in fewer channels. The dimension of the $\mathbb{Y}_1$ block is the number of primitive channels used in the first interval, and the dimension of the $\mathbb{Y}_4$ block is the number of primitive channels in the last interval. The blocks $\mathbb{Y}_2$ and $\mathbb{Y}_3$ are not square anymore.

This procedure finally yields the required result, a log derivative matrix over the interval $[S, B]$.

5.3 Implementation details

In this section the implementation of the quasi-adiabatic method in the DYNA program package is described. The division of the total propagation interval in steps and the calculation of the primitive bases of the steps is done with the routine CALC_INTERVALS. This routine stores the different basis sets to disk, and writes out an input file for the so-called scheduler job. This scheduler makes it possible to parallelize the propagation. The scheduler has two parameters, the maximum number of jobs it may be running at any time. If this number is equal to one, there is no parallelization, if it is larger, say n, there will be at most n jobs running at the same time. Therefore, n should not be chosen larger than the number of processors on the machine. The second parameter is an input file, specifying what jobs should be run, and the command-line parameters for these jobs.

The input file, written by CALC_INTERVALS contains one line per step, to tell the scheduler to start a DO_INTERVAL job for each step. The command line parameters for this DO_INTERVAL job are the begin and endpoint of the step, R_i and R_{i+1}, the value $E_{\max}(R_i)$, the number of the primitive basis set to use, and the tolerance that has to be reached on the over the step, τ_i. The routine DO_INTERVAL calculates the quasi-adiabatic basis of the step, propagates the CC equation over the step, using the routine CAUCHY described in chapter 3, and transforms the resulting LD matrix back into the primitive basis of the step. This LD matrix is then stored to disk.

The routine GLUE reads all the LD matrices of the different steps from disk, and combines them in one LD matrix over the total interval $[S, B]$. After that, the routine JDIST2 [Chap. 3] can be called to combine this LD matrix with the LD matrix over the interval $[A, S]$ and to calculate a scattering or reactance matrix.

Chapter 6

Results of the quasi-adiabatic optimization

In this chapter, the tests of the quasi-adiabatic optimization of the coupled channels algorithm are presented. In the first section, the test system is described. In the second section, the tests and the reference calculations that were performed are described. The results of these tests follow in the last section.

6.1 Test system

The system, used in section 4.5 to test the optimization of the use of the NAG routines, is too small to test the quasi-adiabatic optimization. The calculations are too fast to time the differences accurately.

Therefore, an other test system has been used, namely the reactive collision of hydrogen (H) atoms and oxygen molecules (O_2). The calculations have been performed on the H + O_2 arrangement, for which the external region extends from 5 Bohr (S_1 in figure 1.2) to 20 Bohr (B_1). The total scattering energy (E) was 3.195355 eV, which is 0.187 eV above the H + O_2 well in the potential energy surface. This reaction was studied with the method before the optimizations were made,[9,12] and by other methods.[13] To save computing time in the present reference calculations with the old method, the propagation interval was divided into 16 logarithmically spaced intervals. The number of channels used in each interval was scaled linearly with the scattering coordinate R, starting with the total number of channels in the first interval and dropping to only the asymptotically open channels at $R = B$.

To speed up the present tests, the atomic mass of oxygen is reduced to 0.15 times its normal atomic mass. This effectively results in lowering the scattering energy with respect to the rovibrational levels of O_2, or, looked upon from the other side, increasing the energy of the rovibrational levels with respect to the total energy. This reduces the number of asymptotically open channels from 95 to 15, and also the total number of channels necessary to reach a certain accuracy from about 500 to 225 for the most expensive (and best) calculations performed.

6.2 Description of the tests

First, some calculations with the old method [Chap. 3] were performed. The tolerance on the propagation was set to 10^{-8} in these calculations, and the energy up to which the (primitive) channels were taken into account at the begin point of the propagation interval ($E_{\mathrm{max}}^{\mathrm{prim}}$) varied. These energies, and the number of channels (n_{chan}) that they resulted in, are given in table 6.1. In these calculations with the old method, the optimization of the parameters of the NAG routine D02LAF was used [Chap. 4].

Table 6.1: Energies and number of channels in the reference calculations with the old method

$E_{\mathrm{max}}^{\mathrm{prim}}$ (eV)	n_{chan}	Error estimate
4.675	73	9.55×10^{-3}
5.0	91	1.39×10^{-3}
5.445355	119	3.58×10^{-4}
5.75	139	2.41×10^{-4}
6.0	156	1.43×10^{-4}
6.25	178	1.22×10^{-4}
6.5	202	4.23×10^{-5}
6.75	227	0^{a}

[a] By definition

The error estimates given in table 6.1 are the maxima of the absolute errors in the $\mathbb{S}$ matrix elements, with respect to the results of the most expensive calculation, with 227 channels. From the trend in these accuracy estimates, the error of this 227 channel calculation can be estimated to be of the order of 10^{-5}. This calculation will be used as a reference calculation in the testing of the new method.

With the new method several series of calculations are performed. At first, two series are calculated. In the first series, the maximum energy, up to which *local* channels are included at the begin point of the propagation ($E_{\mathrm{max,S}}$), is varied, while the tolerance parameter is kept constant at 10^{-8}. The value of $E_{\mathrm{max,S}}$ ranges from 5.0 to 6.75 eV, with steps of 0.25 eV. It can be expected that the most expensive calculation in this series is more accurate than the reference calculation with the old method, because here local channels are included up to the same energy as the primitive channels in the reference calculation. To be able to include these local channels, more primitive channels are included in the primitive basis in the first interval. In table 6.2, for each calculation the number of local (n_{local}) and primitive (n_{prim}) channels in the first interval is given. There is also an influence on the step size from the $E_{\mathrm{max}}^{\mathrm{user}}$ parameter. The number of steps (in which one local basis is used) is also given in table 6.2.

Table 6.2: Influence of $E_{\mathrm{max,S}}$ on number of channels and step size.

$E_{\mathrm{max,S}}$	n_{prim}	n_{local}	number of intervals
5.0	155	74	18
5.25	199	89	19
5.5	237	109	20
5.75	252	131	21
6.0	266	154	22
6.25	283	177	23
6.5	264	200	24
6.75	300	224	25

The primitive channels were calculated up to an energy of 7.445355 eV, just as with the reference calculations, in order to make sure that the same primitive basis functions are used, and that the quadrature used in determining the bases of the steps is the same as the quadrature used in the reference calculations. This results in a total number of 320 primitive channels. In the most expensive calculation, 300 of these channels are used.

In the second series, $E_{\mathrm{max,S}}$ is kept constant on 6.75 eV, and the tolerance varies in the range of 10^{-1} to 10^{-8}. The tolerance mostly influences the step size [Table 6.3]. This table shows the number of steps used over the total propagation interval, because the step size is variable over this interval and thus difficult to show in one number.

The influence of the tolerance on the number of channels is somewhat more subtle. It does not influence the number of channels used in the first interval, but the influence is via the step size. This is demonstrated in figure 6.1. In this figure, the number of channels used is plotted against the scattering coordinate. When the tolerance is set lower, the length of the intervals becomes shorter. The $E_{\mathrm{max}}(R)$ used to calculate the number of channels, scales linearly with the scattering coordinate. Because of the smaller intervals with the lower tolerance, the number of channels lies closer to this straight line, i.e. the "stairs" are shorter, and lower. This results in a slight decrease in the number of channels used, when the tolerance becomes lower, for a fixed value of $E_{\mathrm{max,S}}$.

Table 6.3: Influence of tolerance on step size.

tolerance	numb. of intervals
10^{-1}	9
$10^{-1.5}$	10
10^{-2}	10
$10^{-2.5}$	11
10^{-3}	11
10^{-4}	13
10^{-5}	14
10^{-6}	17
10^{-7}	20
10^{-8}	25

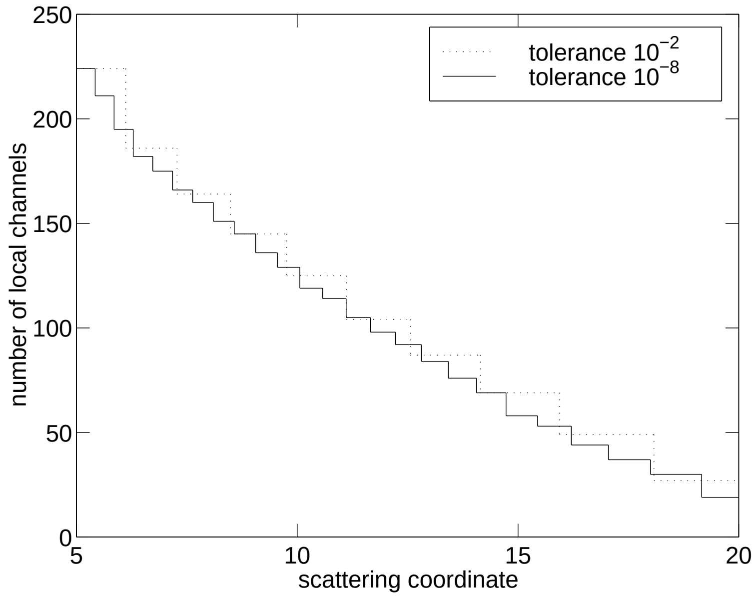

Figure 6.1: The net influence of the tolerance on the number of channels used.

The effect of the $E_{\mathrm{max,S}}$ parameter on the results of these calculations can be understood very well. The effect of the tolerance parameter is seemingly more difficult to understand. Therefore some other calculations are performed, with $E_{\mathrm{max,S}} = 5.75$ eV, and a tolerance varying from 10^{-1} to 10^{-10}.

6.3 Results of the tests

As a measure of accuracy of the calculations, the accuracy of the constructed inelastic $\mathbb{S}$ matrix is used. The error in this matrix is defined to be

$$\text{error} = \max_{1 \leq i,j \leq n_{\mathrm{open}}} |\mathbb{S}_{ij} - \mathbb{S}_{ij}^{\mathrm{ref}}| \tag{6.1}$$

where n_{open} is the number of open channels and the dimension of the $\mathbb{S}$ matrix. The result of the reference calculation is called $\mathbb{S}^{\mathrm{ref}}$. Two reference calculations are used. The first is the most expensive calculation with the old method (227 channels) This reference calculation will be labeled "prim", from "primitive". The second one is the most expensive calculation with the new method ($E_{\mathrm{max,S}} = 6.75$ eV and tolerance $= 10^{-8}$), this one will be labeled with "qa", from "quasi-adiabatic".

In appendix B, all the timing and accuracy results are listed for all the calculations performed. In figure 6.2, the errors of the first test series (with varying $E_{\mathrm{max,S}}$ parameter) is plotted, using the two reference calculations. The error in the "prim" reference calculation is about 3×10^{-5}, and the more expensive quasi-adiabatic calculations are better, as expected. It is seen, that the logarithm of the error scales linearly with the $E_{\mathrm{max,S}}$ parameter, which is as expected. Including more channels should give a more accurate answer.

In figure 6.3, the errors of the series with varying tolerance are plotted against this tolerance. This plot shows the effect of the tolerance parameter for $E_{\mathrm{max,S}} = 6.75$ eV and for $E_{\mathrm{max,S}} = 5.75$ eV. First we have tried to explain the results for the $E_{\mathrm{max,S}} = 6.75$ eV calculations. In the tests with tolerance $10^{-1.5}$ and 10^{-1} (and $E_{\mathrm{max,S}} = 6.75$), the tolerance passed to the NAG routine [Eqs. (4.6) and (4.7)] becomes larger than 1, and it is reset to 1. As a result, the results of these calculations are more accurate than expected. This explains the down going part of the curve.

The first part of the curve is steep (tolerance $\geq 10^{-2}$), as expected [Chap. 5]. With smaller tolerance, the curve levels out. This could be explained by the error due to the almost fixed basis size (fixed $E_{\mathrm{max,S}}$), but it is still possible to improve the results with a smaller value of the tolerance. The calculation with $E_{\mathrm{max,S}} = 5.75$ and tolerance 10^{-9} is as accurate as the calculation with $E_{\mathrm{max,S}} = 6.75$ and tolerance 10^{-4}. This can be easily understood when realizing that a smaller value of the tolerance

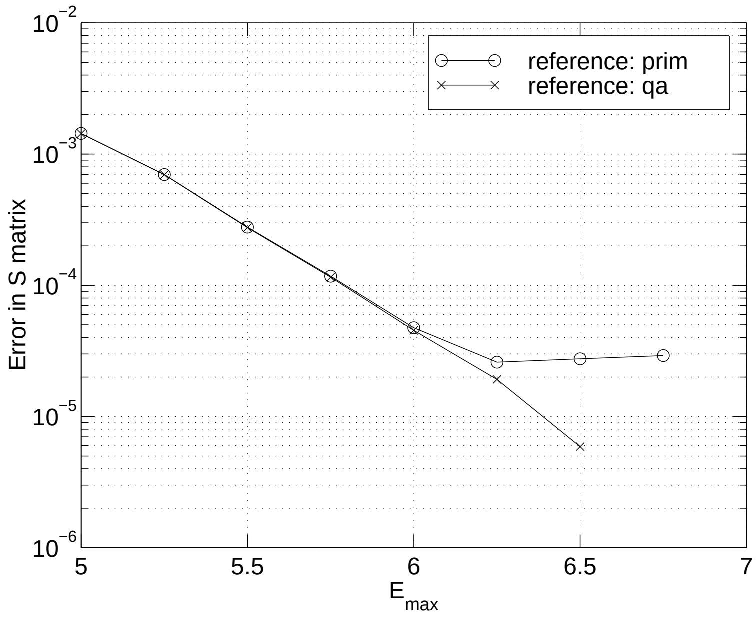

Figure 6.2: Errors in the $\mathbb{S}$ matrices from the $E_{\max}$ test series.

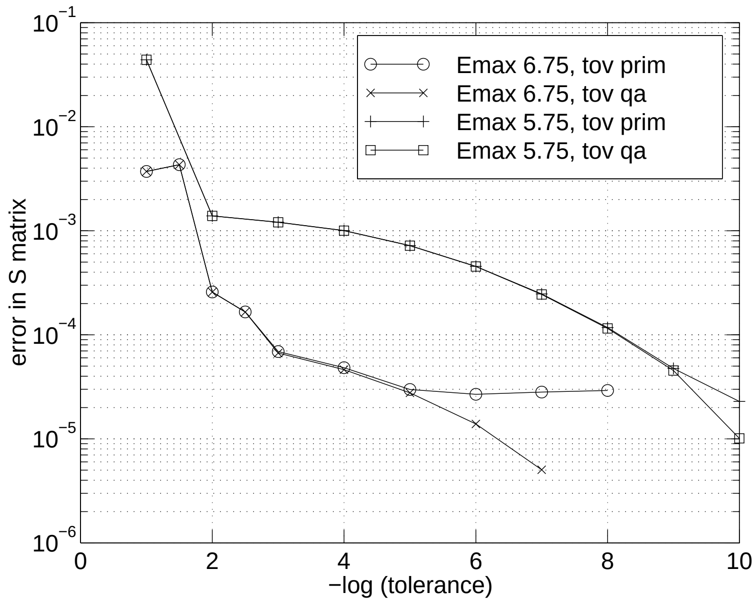

Figure 6.3: Errors in the $\mathbb{S}$ matrices from the tolerance test series.

gives a smaller step size. One local basis is used over a shorter interval, the local basis is better adapted to the potential.

To test this hypothesis, a calculation has been performed with the intervals and quasi-adiabatic bases from the calculation with $E_{\mathrm{max,S}} = 5.75$ eV and tolerance 10^{-10}. The tolerance, given to the NAG routine, has been set four orders of magnitude higher. The results of these two calculations are the same up to 8 significant figures, where the error in the $\mathbb{S}$ matrix is of the order of 10^{-5}. This proves the hypothesis, that the NAG propagation is almost perfect already at relatively low tolerances of 10^{-3}, or 10^{-4}, and that the tolerance parameter has the main influence on the step size.

This perfectness can be explained by the fact that the NAG routine demands that the (relative or absolute) error in all the solutions is smaller that the required tolerance. So in fact, the error of the solution which grows the most exponentially (the most strongly closed channel) is controlled. The other channels, with in absolute value smaller k values, are propagated with the same step size, the results for these channels are thus more accurate than expected.

From this results [Figs. 6.2 and 6.3], it can be concluded that the error in the result can be predicted by the setting of the two parameters $E_{\mathrm{max,S}}$ and tolerance. The tolerance has not very much influence on the accuracy of the propagation, it might be a further optimization to set the NAG tolerance to a fixed value of 10^{-4}, and to use the present tolerance parameter only to control the number of quasi-adiabatic intervals. The next thing that has to be investigated, is whether the second aim of the project has also succeeded. Did the method become faster?

Figure 6.4 is a plot of the user time that the calculations took, versus the error in the result. Each dot represents one test calculation. Three of the quasi-adiabatic points fall off the line. These three points represent the two least accurate calculations of the tolerance series with $E_{\mathrm{max,S}} = 6.75$ eV and the least accurate one from the $E_{\mathrm{max,S}} = 5.75$ eV series. It appears that there is some minimum number of user time needed for the propagation, and that this minimum time depends on the parameter setting for $E_{\mathrm{max,S}}$. In this picture it can be seen clearly, that it can be very inefficient to use a "wrong" combination of parameter settings.

One of the quasi adiabatic calculations has been marked with an "a)" in figure 6.4. This is the test calculation, in which the tolerance for the NAG was 4 orders higher than the tolerance used to calculate the intervals and quasi-adiabatic bases. This calculation shows that it is possible to optimize this new method further.

It can also be seen that the quasi-adiabatic method is faster than the old method, in particular when a higher accuracy is required. The scaling of the computing time needed with the required accuracy is also more favorable, i.e. the increase in computer time needed to gain an order of magnitude in accuracy is less for the quasi-adiabatic than for the old method.

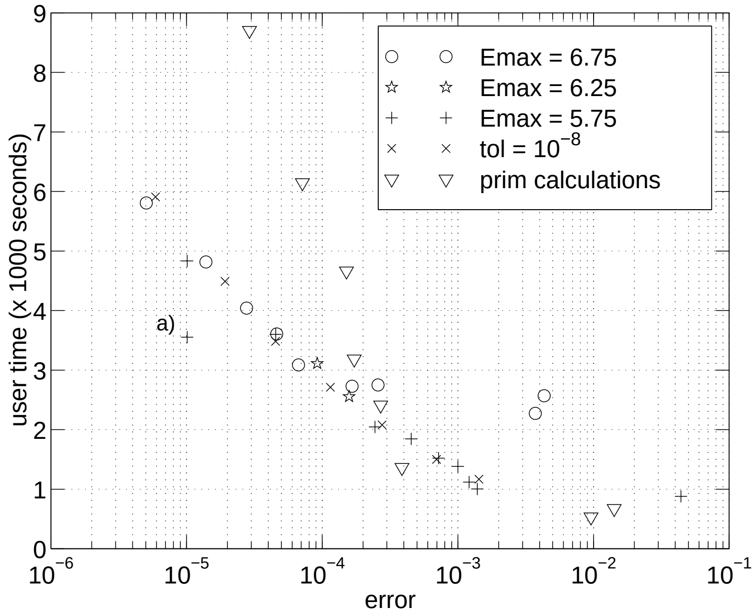

Figure 6.4: User time versus error for all test calculations

For a typical calculation, an error of 10^{-5} or 10^{-4} is acceptable. In this accuracy region, the quasi-adiabatic method is approximately a factor two faster that the reference calculations performed with the old method. In these reference calculations, the optimization in the use of the NAG routines [Chap. 4] has already been used, which gave a speedup of about 30%. Together, the two optimizations result in a speedup of about a factor three.

A further remark about this result is, that the reference calculations used an ad hoc reduction of the number of channels when the propagation proceeded outward. This reduction worked fine, but was never tested thoroughly. The fairest comparison would be with the old method, in which this reduction has not been used. Then the gain in speed would be even much larger.

Chapter 7

Conclusions and summary

In this report, the optimization of the coupled channels algorithm, as implemented in the Nijmegen DYNA program package,[1] has been studied. Coupled channels can be used to solve inelastic scattering problems, or to construct distorted waves that can be used in a variational reactive scattering calculation, the main purpose of DYNA.

The optimization consists of two steps. The coupled channels algorithm uses a NAG library routine to solve numerically a system of second order differential equations, with given initial conditions. In the first step, the use of this routine was optimized. This resulted in two formulae [Eqs. (4.6) and (4.7)], to calculate the optimal step size and tolerance parameter settings for this NAG routine. With these formulae, it is possible to use the NAG routine to solve the differential equations up to a certain accuracy with the least computing effort. This resulted in a reduction of computing time with about 30%.

The second step involved the design of a new algorithm to solve the coupled channels equation. The idea is to use a locally optimal basis set to propagate the CC equation. To achieve this, several partial algorithms had to be designed. The first algorithm is neccessary to calculate the optimal division of the total integration interval in steps, in which one local basis is used. The calculation of each basis requires evaluation and diagonalization of a large potential matrix, which is very time consuming. If the steps are too small, the calculation of the basis sets takes very much time. But on the other hand, if the steps are too large, the basis is not optimal over the whole step any more, which also gives rise to more computing effort.

The second algorithm is used to solve the CC equations over one step, in the local basis, and then transform the results into the final basis. The last algorithm that has been developed, is used to combine the results of the different steps.

From tests, it can be concluded that this new method is particularly efficient when high-accuracy results are required. The new quasi-adiabatic method is about 50% faster than the old method. The scaling of computing time with accuracy is also more favorable. This means that the growth in computing time when more accurate results are required, is less than with the old method.

The two optimizations together give a reduction in computing time of about a factor three. Furthermore the new method only requires a minimal number of parameters to describe the desired accuracy in the propagation. The values for all the other parameters are determined automatically. This new method has been implemented as a black box routine in the DYNA program package. The user only has to supply two parameters to describe the propagation of the coupled channels equation.

Dankwoord

Als eerste wil ik mijn stage begeleider, Gerrit Groenenboom bedanken voor alle heldere uitleg en begeleiding, en omdat hij altijd tijd had om me met problemen te helpen.

Verder wil ik Ad van der Avoird bedanken voor de dagelijkse supervisie op de afdeling. Paul Wormer wil ik bedanken voor het kritisch doorlezen van het verslag en de lessen over groepentheorie, en Ine Meyer wil ik bedanken, omdat ze er altijd was om probleempjes op te lossen.

De aio's van de afdeling, Tino Heijmen en Michel Geleijns wil ik bedanken voor de wijze raad die zij gevraagd, en vooral ook ongevraagd gaven. Mijn twee medestudenten op de afdeling, Paul van Beurden en Marianne Reedijk, wil ik graag bedanken voor alle lol die we vooral hadden tijdens onze discussies over voetbal en feminisme.

Als laatste wil ik mijn vriend Erik Verbruggen bedanken voor de morele steun, op de paar momentjes dat ik het niet meer helemaal zag zitten, en de liefde die hij me altijd geeft.

Bibliography

[1] The reactive scattering program package DYNA. Developped at the University of Nijmegen by G.C. Groenenboom. Institute of Theoretical Chemistry, Toernooiveld 1, 6525 ED Nijmegen.

[2] *The NAG Fortran Library*, available from the Numeriacal Algorithms Group Limited, Wilkinson House, Jordan Hill Road, Oxford, United Kingdom.

[3] D. Secrest, *Rotational Excitation I: The Quantal Treatment*, Chap. 8 from *Atom-molecule collision theory*, ed. Richard R. Bernstein, Plenum Press, New York and London (1979).

[4] D.E. Manolopoulos, *An improved log derivative method for inelastic scattering*, J. Chem. Phys. **85**, 6425-6429 (1986)

[5] M.H. Alexander, *Hybrid quantum scattering algorithms for long-range potentials*, J. Chem. Phys. **81**, 4510-4516 (1984)

[6] M.H. Alexander and D.E. Manolopoulos, *A stable linear reference potential algorithm for solution of the quantum close-coupled equations in molecular scattering theory*, J. Chem. Phys. **86**, 2044-2050 (1987)

[7] E.B. Stechel, R.B. Walker, and J.C. Light, *R-matrix solution of coupled equations for inelastic scattering*, J. Chem. Phys. **69**, 3518-3531 (1978).

[8] D. Secrest and B.R. Johnson, *Exact Quantum-Mechanical Calculation of a Collinear Collision of a Particle with a Harmonic Oscillator*, J. Chem. Phys. **45**, 4556-4570 (1966).

[9] G.C. Groenenboom, *Discrete variational quantum reactive scattering method with optimal distorted waves. I. Theory*, J. Chem. Phys. **108**, 5670-5676, (1998)

[10] G.C. Groenenboom and D.T. Colbert, *Combining the discrete variable representation with the S-matrix Kohn method for quantum reactive scattering*, J. Chem. Phys. **99**, 9681-9696, (1993)

[11] B.R. Johnson, National Resource for Computation in Chemistry, University of California, Berkeley, *The Log-Derivative and Renormalized Numerov Algorithms*, NRCC Proceedings **5**, 86-104, (1979)

[12] G.C. Groenenboom, *Discrete variational quantum reactive scattering method with optimal distorted waves. II. Application to the reaction H + $O_2 \rightarrow OH + O$*, J. Chem. Phys. **108**, 5677-5682, (1998)

[13] R.T. Pack, E.A. Butcher, and G.A. Parker, *Accurate three-dimensional quantum probabilities and collision lifetimes of the H + O_2 combustion reaction*, J. Chem. Phys. **102**, 5998-6012, (1995)

Appendix A

Specification of the NAG routines used

In this appendix the precise specification of the NAG routines, used in the coupled channels algorithm is given.

A.1 D02LAF

```
      SUBROUTINE D02LAF (FCN, N, XBEGIN, XEND, Y, YP, YDP,
     .                   RWORK, LRWORK, IFAIL)

      INTEGER   N, LRWORK, IFAIL
      REAL*8    XBEGIN, XEND, Y(N), YP(N), YDP(N),
     .          RWORK(LRWORK)
      EXTERNAL  FCN
```

The meaning of the parameters is:

- `FCN` The subroutine, supplied by the user. This routine has to evaluate the functions f_i, for given values of the arguments. Its specification looks:

  ```
  SUBROUTINE FCN (N, X, Y, F)

  INTEGER N
  REAL*8  X, Y(N), F(N)
  ```

  The parameters are

  - `N` Input parameter. The number of equations n.
  - `X` Input parameter. Value of the integration variable x.

  - `Y(N)` Input parameter. Array with the values y_i in the point x.
  - `F(N)` Output parameter. Array with the values of the second derivatives f_i.

- `N` The number of equations.
- `XBEGIN` Input: the starting point of the integration range. On output the value of `XBEGIN` has been changed to the new value of the integration variable x. When operating in interval mode, and when the integration succeeds, this value will be `XEND`.
- `Y(N)` Input: the initial values of the solution $y_1, \ldots, y_n$. On output: the values of the solution at the exit value of `XBEGIN`.
- `YP(N)` Input: the initial values of the derivative of the solution y'_i . On output: the computed values of these derivatives at the exit value of `XBEGIN`
- `YDP(N)` Output: the computed values of the second derivatives f_i at the exit value of `XBEGIN`
- `RWORK(LRWORK)` Workspace for D02LAF.
- `LRWORK` The declared dimension of the workspace array.
- `IFAIL` Error control parameter. All the NAG routines use this parameter. For a description of its use we refer to the NAG manual.[2]

A.2 d02lxf

```
 SUBROUTINE D02LXF (N, H, TOL, THRES, THRESP, MAXSTP,
.                   START, ONESTP, HIGH, RWORK, LRWORK,
.                   IFAIL)

 INTEGER    N, MAXSTP, LRWORK, IFAIL
 REAL*8     H, TOL, THRES(N), THRESP(N), RWORK(LRWORK)
 LOGICAL    START, ONESTP, HIGH
```

The meaning of the parameters is:

- `N` Number of equations to be solved.
- `H` Input parameter. `H` may specify a hint for the next step size to be attempted by D02LAF.
- `TOL` Input parameter to control the relative error in the solution.

- `THRES(N)` and `THRESP(N)` Input parameters with threshold values for use in the error control.
- `MAXSTP` Input parameter. Gives a bound to the number of steps attempted by D02LAF.
- `START` Input: indicates whether the next call to D02LAF is for a new problem or not. `START` = `.FALSE.` indicates that the call to D02LXF is prior to a continuation call of D02LAF, for example to change some of the options. On output: `START` = `.FALSE.`
- `ONESTP` Indicates whether D02LAF will be used in step mode (`ONESTP` = `.TRUE.`) or in interval mode. Step mode means that D02LAF will return after each succeeded step, and interval mode means that D02LAF returns only at the end of the integration interval.
- `HIGH` Indicates whether the high or low order formula pair should be used. The high order method uses a 12(10) formula pair, and the lower order method uses a 6(4) formula pair.
- `RWORK(LRWORK)` Workspace for D02LAF. This array is used to pass information from D02LXF to D02LAF, so it must be the same array as used in the call to D02LAF, and its contents must not be changed.
- `LRWORK` Declared dimension of the array `RWORK`. There is a constraint on the minimum size of this workspace array.
- `IFAIL` Error control parameter.

Appendix B

Test results

In the following tables, the results of all the tests of the quasi-adiabatic method are listed. For each calculation, two measures of the accuracy are given, namely the error with respect to the “prim” (Error Prim) and “qa” (Error QA) reference calculations, as described in section 6.3. Furthermore is a measure of time, needed for the propagation, given. This measure is the user time (in seconds), the propagation took on a SUN Ultra Sparc machine, with 250 MHz processors. No parallelization was used.

n_{chan}	Error QA	Error Prim	Time
73	9.58×10^{-3}	9.55×10^{-3}	533
91	1.42×10^{-3}	1.39×10^{-3}	673
119	3.87×10^{-4}	3.58×10^{-4}	1364
139	2.70×10^{-4}	2.41×10^{-4}	2411
156	1.72×10^{-4}	1.43×10^{-4}	3185
178	1.51×10^{-4}	1.22×10^{-4}	4663
202	7.15×10^{-5}	4.23×10^{-5}	6146
227[a]	2.91×10^{-5}	0[b]	8703

[a] “prim” reference calculation
[b] by definition

Table B.1: Accuracy and timing results for the old method test calculations.

$E_{\max}$	tolerance	Error QA	Error Prim	Time
5.0	10^{-8}	1.43×10^{-3}	1.44×10^{-3}	1167
5.25	10^{-8}	6.96×10^{-4}	6.98×10^{-4}	1500
5.5	10^{-8}	2.76×10^{-4}	2.78×10^{-4}	2080
5.75	10^{-8}	1.15×10^{-4}	1.17×10^{-4}	2712
6.0	10^{-8}	4.53×10^{-5}	4.75×10^{-5}	3482
6.25	10^{-8}	1.92×10^{-5}	2.59×10^{-5}	4490
6.5	10^{-8}	5.90×10^{-6}	2.65×10^{-5}	5910
6.75^a	10^{-8}	0^b	2.91×10^{-5}	7203

^{a} "qa" reference calculation
^{b} by definition

Table B.2: Accuracy and timing results for quasi-adiabatic test series with tolerance 10^{-8} and varying $E_{\max}$

$E_{\max}$	tolerance	Error QA	Error Prim	Time
6.75	10^{-1}	3.72×10^{-3}	3.72×10^{-3}	2273
6.75	$10^{-1.5}$	4.33×10^{-3}	4.33×10^{-3}	2569
6.75	10^{-2}	2.58×10^{-4}	2.58×10^{-4}	2750
6.75	$10^{-2.5}$	1.66×10^{-4}	1.66×10^{-4}	2728
6.75	10^{-3}	6.69×10^{-5}	6.90×10^{-5}	3087
6.75	10^{-4}	4.62×10^{-5}	4.83×10^{-5}	3607
6.75	10^{-5}	2.77×10^{-5}	2.99×10^{-5}	4042
6.75	10^{-6}	1.39×10^{-5}	2.68×10^{-5}	4816
6.75	10^{-7}	5.05×10^{-6}	2.82×10^{-5}	5807

Table B.3: Accuracy and timing results for quasi-adiabatic test series with varying tolerance and $E_{\max} = 6.75$ eV

$E_{\max}$	tolerance	Error QA	Error Prim	Time
6.25	10^{-5}	1.58×10^{-4}	1.60×10^{-4}	2555
6.25	10^{-6}	9.19×10^{-5}	9.41×10^{-5}	3110

Table B.4: Accuracy and timing results for quasi-adiabatic tests with varying tolerance and $E_{\max} = 6.25$ eV

$E_{\max}$	tolerance	Error QA	Error Prim	Time
5.75	10^{-1}	4.41×10^{-2}	4.41×10^{-2}	880
5.75	10^{-2}	1.39×10^{-3}	1.39×10^{-3}	1004
5.75	10^{-3}	1.21×10^{-3}	1.21×10^{-3}	1118
5.75	10^{-4}	1.00×10^{-3}	1.01×10^{-3}	1382
5.75	10^{-5}	7.20×10^{-4}	7.22×10^{-4}	1533
5.75	10^{-6}	4.53×10^{-4}	4.56×10^{-4}	1844
5.75	10^{-7}	2.45×10^{-4}	2.47×10^{-4}	2045
5.75	10^{-9}	4.54×10^{-5}	4.75×10^{-5}	3599
5.75	10^{-10}	1.01×10^{-5}	2.29×10^{-5}	4833

Table B.5: Accuracy and timing results for quasi-adiabatic test series with varying tolerance and $E_{\max} = 5.75$ eV