CHAPTER 4

Computational and Numerical Details

This chapter consists of 3 parts. The first part is the description of the numerical solution of the LS equations. The second part will be the discussion on the difficulties encountered during the CC and CCO calculations. Finally, the last part consists of the numerical convergence test. The CC, CCO and UBA calculations are done using the computer code developed by Mitroy (Mitroy (1993a, 1996)).

4.1   Numerical solutions of Lippmann-Schwinger Equations

In order to assist our discussion, we need to abbreviate the T-matrix and V-matrix of the partial-wave form LS equations (equation (2.60)):

$$T^{(J)}_{\alpha' L' \alpha L} = T_{\alpha' \alpha} \qquad\qquad V^{(J)}_{\alpha' L' \alpha L} = V_{\alpha' \alpha} \tag{4.1a}$$

$$T^{(J)}_{\beta' L' \alpha L} = T_{\beta' \alpha} \qquad\qquad V^{(J)}_{\beta' L' \alpha L} = V_{\beta' \alpha} \tag{4.1b}$$

So, equation (2.60) can be rewritten as:

$$\mathbf{T}_{\alpha'\alpha}(q,k_\alpha) = \mathbf{V}_{\alpha'\alpha}(q,k_\alpha) + \sum_{\alpha''}\int_0^\infty dq' q'^2 \, \mathbf{V}_{\alpha'\alpha''}(q,q')\mathbf{G}_{\alpha''}(q'^2)\mathbf{T}_{\alpha''\alpha}(q',k_\alpha)$$
$$+ \sum_{\beta''}\int_0^\infty dq' q'^2 \, \mathbf{V}_{\alpha'\beta''}(q,q')\mathbf{G}_{\beta''}(q'^2)\mathbf{T}_{\beta''\alpha}(q',k_\alpha) \tag{4.2a}$$

$$\mathbf{T}_{\beta'\alpha}(q,k_\alpha) = \mathbf{V}_{\beta'\alpha}(q,k_\alpha) + \sum_{\alpha''}\int_0^\infty dq' q'^2 \, \mathbf{V}_{\beta'\alpha''}(q,q')\mathbf{G}_{\alpha''}(q'^2)\mathbf{T}_{\alpha''\alpha}(q',k_\alpha)$$
$$+ \sum_{\beta''}\int_0^\infty dq' q'^2 \, \mathbf{V}_{\beta'\beta''}(q,q')\mathbf{G}_{\beta''}(q'^2)\mathbf{T}_{\beta''\alpha}(q',k_\alpha) \tag{4.2b}$$

In equation (4.2), the 2 different Green's functions are defined as:

$$G_{\alpha''}(q^2) = \Re\left[\frac{1}{\frac{1}{2}(k^2_{\alpha''} - q^2)}\right] - \frac{i\pi}{k_{\alpha''}}\delta(k_{\alpha''} - q) \tag{4.3a}$$

$$G_{\beta''}(q^2) = \Re\left[\frac{1}{\frac{1}{2}\left(k_{\beta''}^2 - q^2\right)}\right] - \frac{i\pi}{k_{\beta''}}\delta(k_{\beta''} - q) \qquad (4.3b)$$

The term $\Re$ is the principal value of the Green's functions. The term q is an off shell momentum and it can take any value. The terms $k_{\alpha''}^2$ and $k_{\beta''}^2$ are the on shell momenta of the target electrons for channel α'' and the Ps formation for channel β'', respectively:

$$k_{\alpha''}^2 = 2(E - \epsilon_{\alpha''}) \qquad\qquad k_{\beta''}^2 = 4(E - \epsilon_{\beta''}) \qquad (4.4)$$

where the terms $\epsilon_{\alpha''}$ and $\epsilon_{\beta''}$ correspond to:

$$\epsilon_{\alpha''} = \varepsilon_{core} + \varepsilon_{\alpha''} \qquad\qquad \epsilon_{\beta''} = \varepsilon_{core} + \varepsilon_{\beta''} \qquad (4.5)$$

The Gaussian quadrature method is used to solve the equations. In order to apply this method, we transform the LS equations into an algebraic form and the equations can be expressed in term of the coordinate x_{gn} which is defined as:

$$x_{gn} = \begin{cases} k_g & where \;\; n = 1 \\ q_{n-1} & where \;\; n = 2,3,\ldots,N+1 \end{cases} \qquad (4.6)$$

where the subscript term g indicates the type of Green's function, i.e. $g = \alpha''$ for Green's function for channel α'' and $g = \beta''$ for Green's function for channel β''. After defining the coordinate x_{gn}, the potential matrices ($V_{\alpha'\alpha''}(x_{\alpha'n}, x_{\alpha''n})$, $V_{\alpha'\beta''}(x_{\alpha'n}, x_{\beta''n})$, $V_{\beta'\alpha''}(x_{\beta'n}, x_{\alpha''n})$, $V_{\beta'\beta''}(x_{\beta'n}, x_{\beta''n})$) and solution vectors $(T_{\alpha'\alpha}(x_{\alpha'n}, k_\alpha), T_{\beta'\alpha}(x_{\beta'n}, k_\alpha))$ can be written in coordinate form as well.

By using the relations below, the singularities caused by the Green's function at $q = k_{\alpha''}$ and $q = k_{\beta''}$ can be removed:

$$\Re\int_0^\infty dq \left[\frac{q^2}{\frac{1}{2}\left(k_{\alpha''}^2 - q^2\right)}\right] = 0 \qquad (4.7a)$$

$$\Re \int_0^\infty dq \left[\frac{q^2}{\frac{1}{2}\left(k_{\beta''}^2 - q^2\right)}\right] = 0 \tag{4.7b}$$

Thus, the LS equations are:

$$
\begin{aligned}
\mathbf{T}_{\alpha'\alpha}\left(x_{\alpha'n}, k_\alpha\right) &= \mathbf{V}_{\alpha'\alpha}\left(x_{\alpha'n}, k_\alpha\right) \\
&+ \sum_{\alpha''}\sum_{n'=2}^{N+1} w_{n'-1}\left[\frac{x_{\alpha''n'}^2 \mathbf{V}_{\alpha'\alpha''}\left(x_{\alpha'n}, x_{\alpha''n'}\right)\mathbf{T}_{\alpha''\alpha}\left(x_{\alpha''n'}, k_\alpha\right) - k_{\alpha''}^2 \mathbf{V}_{\alpha'\alpha''}\left(x_{\alpha'n}, k_{\alpha''}\right)\mathbf{T}_{\alpha''\alpha}\left(k_{\alpha''}, k_\alpha\right)}{\frac{1}{2}\left(k_{\alpha''}^2 - x_{\alpha''n'}^2\right)}\right] \\
&- i\pi k_{\alpha''}\mathbf{V}_{\alpha'\alpha''}\left(x_{\alpha'n}, k_{\alpha''}\right)\mathbf{T}_{\alpha''\alpha}\left(k_{\alpha''}, k_\alpha\right) \\
&+ \sum_{\beta''}\sum_{n'=2}^{N+1} w_{n'-1}\left[\frac{x_{\beta''n'}^2 \mathbf{V}_{\alpha'\beta''}\left(x_{\alpha'n}, x_{\beta''n'}\right)\mathbf{T}_{\beta''\alpha}\left(x_{\beta''n'}, k_\alpha\right) - k_{\beta''}^2 \mathbf{V}_{\alpha'\beta''}\left(x_{\alpha'n}, k_{\beta''}\right)\mathbf{T}_{\beta''\alpha}\left(k_{\beta''}, k_\alpha\right)}{\frac{1}{4}\left(k_{\beta''}^2 - x_{\beta''n'}^2\right)}\right] \\
&- i\pi k_{\beta''}\mathbf{V}_{\alpha'\beta''}\left(x_{\alpha'n}, k_{\beta''}\right)\mathbf{T}_{\beta''\alpha}\left(k_{\beta''}, k_\alpha\right)
\end{aligned} \tag{4.8a}
$$

$$
\begin{aligned}
\mathbf{T}_{\beta'\alpha}\left(x_{\beta'n}, k_\alpha\right) &= \mathbf{V}_{\beta'\alpha}\left(x_{\beta'n}, k_\alpha\right) \\
&+ \sum_{\alpha''}\sum_{n'=2}^{N+1} w_{n'-1}\left[\frac{x_{\alpha''n'}^2 \mathbf{V}_{\beta'\alpha''}\left(x_{\beta'n}, x_{\alpha''n'}\right)\mathbf{T}_{\alpha''\alpha}\left(x_{\alpha''n'}, k_\alpha\right) - k_{\alpha''}^2 \mathbf{V}_{\beta'\alpha''}\left(x_{\beta'n}, k_{\alpha''}\right)\mathbf{T}_{\alpha''\alpha}\left(k_{\alpha''}, k_\alpha\right)}{\frac{1}{2}\left(k_{\alpha''}^2 - x_{\alpha''n'}^2\right)}\right] \\
&- i\pi k_{\alpha''}\mathbf{V}_{\beta'\alpha''}\left(x_{\beta'n}, k_{\alpha''}\right)\mathbf{T}_{\alpha''\alpha}\left(k_{\alpha''}, k_\alpha\right) \\
&+ \sum_{\beta''}\sum_{n'=2}^{N+1} w_{n'-1}\left[\frac{x_{\beta''n'}^2 \mathbf{V}_{\alpha'\beta''}\left(x_{\beta'n}, x_{\beta''n'}\right)\mathbf{T}_{\beta''\alpha}\left(x_{\beta''n'}, k_\alpha\right) - k_{\beta''}^2 \mathbf{V}_{\beta'\beta''}\left(x_{\beta'n}, k_{\beta''}\right)\mathbf{T}_{\beta''\alpha}\left(k_{\beta''}, k_\alpha\right)}{\frac{1}{4}\left(k_{\beta''}^2 - x_{\beta''n'}^2\right)}\right] \\
&- i\pi k_{\beta''}\mathbf{V}_{\beta'\beta''}\left(x_{\beta'n}, k_{\beta''}\right)\mathbf{T}_{\beta''\alpha}\left(k_{\beta''}, k_\alpha\right)
\end{aligned} \tag{4.8b}
$$

Defining 4 kernels:

$$K_{\alpha'\alpha''}\left(x_{\alpha'n}, x_{\alpha''n}\right) = W_{\alpha''n}V_{\alpha'\alpha''}\left(x_{\alpha'n}, x_{\alpha''n}\right) \tag{4.9a}$$

$$K_{\alpha'\beta''}\left(x_{\alpha'n}, x_{\beta''n}\right) = W_{\beta''n}V_{\alpha'\beta''}\left(x_{\alpha'n}, x_{\beta''n}\right) \tag{4.9b}$$

$$K_{\beta'\alpha''}\left(x_{\beta'n}, x_{\alpha''n}\right) = W_{\alpha''n}V_{\beta'\alpha''}\left(x_{\beta'n}, x_{\alpha''n}\right) \tag{4.9c}$$

$$K_{\beta'\beta''}\left(x_{\beta'n}, x_{\beta''n}\right) = W_{\beta''n}V_{\beta'\beta''}\left(x_{\beta'n}, x_{\beta''n}\right) \tag{4.9d}$$

where $W_{\alpha''n}$ and $W_{\beta''n}$ are the superweights which are represented by:

$$W_{\alpha''n} = \begin{cases} x_{\alpha''n}^2 w_{n-1}\left[\frac{1}{2}\left(k_{\alpha''}^2 - x_{\alpha''n}^2\right)\right]^{-1} & \text{when } n = 2,3,\dots,N+1 \\ -k_{\alpha''}^2 \sum_{n'=1}^{N+1} w_{n'-1}\left[\frac{1}{2}\left(k_{\alpha''}^2 - x_{\alpha''n}^2\right)\right]^{-1} - i\pi k_{\alpha''n'} & \text{when } n = 1 \end{cases}$$

$$\tag{4.10a}$$

$$W_{\beta''n} = \begin{cases} x^2_{\beta''n} w_{n-1} \left[\frac{1}{2} \left(k^2_{\beta''} - x^2_{\beta''n} \right) \right]^{-1} & \text{when } n = 2,3,\dots,N+1 \\ -k^2_{\beta''} \sum_{n'=1}^{N+1} w_{n'-1} \left[\frac{1}{2} \left(k^2_{\beta''} - x^2_{\beta''n} \right) \right]^{-1} - i\pi k_{\beta''n'} & \text{when } n = 1 \end{cases}$$

$$(4.10b)$$

Thus, the final form of equation (4.8) can be written as:

$$\mathbf{T}_{\alpha'\alpha}(x_{\alpha'n}, k_\alpha) = \mathbf{V}_{\alpha'\alpha}(x_{\alpha'n}, k_\alpha) + \sum_{\alpha''} \sum_{n'=1}^{N+1} \mathbf{K}_{\alpha'\alpha''}(x_{\alpha'n}, x_{\alpha''n'}) \mathbf{T}_{\alpha''\alpha}(k_{\alpha''n'}, k_\alpha)$$
$$+ \sum_{\beta''} \sum_{n'=1}^{N+1} \mathbf{K}_{\alpha'\beta''}(x_{\alpha'n}, x_{\beta''n'}) \mathbf{T}_{\beta''\alpha}(k_{\beta''n'}, k_\alpha)$$

$$(4.11a)$$

$$\mathbf{T}_{\alpha'\alpha}(x_{\alpha'n}, k_\alpha) = \mathbf{V}_{\alpha'\alpha}(x_{\alpha'n}, k_\alpha) + \sum_{\alpha''} \sum_{n'=1}^{N+1} \mathbf{K}_{\alpha'\alpha''}(x_{\alpha'n}, x_{\alpha''n'}) \mathbf{T}_{\alpha''\alpha}(k_{\alpha''n'}, k_\alpha)$$
$$+ \sum_{\beta''} \sum_{n'=1}^{N+1} \mathbf{K}_{\alpha'\beta''}(x_{\alpha'n}, x_{\beta''n'}) \mathbf{T}_{\beta''\alpha}(k_{\beta''n'}, k_\alpha)$$

$$(4.11b)$$

4.2   Numerical Details in Electron-Rb Scattering Calculations

Until the present work, the CC, CCO and UBA calculations had only been used for electron and positron scattering from lighter atoms such as sodium and lithium. We encountered a number of challenges in attempting this work for a large atomic system such as Rb. In the following sections, we describe and discuss various aspects of the numerics.

4.2.1 Quadrature Points

Solving the LS equations using Gaussian quadrature has been well detailed by McCarthy and Stelbovics (1983a). Amongst the biggest challenges is the distribution of the quadrature points in the calculations. This aspect becomes more taxing in the two-centre calculations that are solved in the positron scattering, as explained by Mitroy (1993a) and Ratnavelu *et al.* (1996).

In the electron-Rb scattering case, we apply the single Gaussian mesh used by McCarthy and Stelbovics (1983a) (which we will refer to as the bunching transformation method or Method A). Following McCarthy and Stelbovics (1983a), the integration mesh of the coupled channel equation must cover small and large k in order to account for the diffuse structure for monopole transitions. It must cover the closely spaced points near the on-shell values of k to represent the detail of dipole as well. Let a set of Gaussian quadrature with points u_j and weights w_j be defined on the interval [0,1]. A standard conformal transformation for points z defined in the interval [0,1] to points k defined on the interval [0,∞] is:

$$k = \frac{az}{1-z} \tag{4.12}$$

where parameter a is the scale parameter and z is defined as:

$$z = \frac{(u-b_1)^3 + (u-b_2)^3 + b_1^3 + b_2^3}{(1-b_1)^3 + (1-b_2)^3 + b_1^3 + b_2^3} \tag{4.13}$$

where $b_1 = 0.5 + b$ and $b_2 = 0.5 - b$. The values of the bunching parameter, b must be less than 1 which will cluster the points around a.

Table 4.1 shows 3 sets of quadrature points generated by varying the scale parameter a with fixed bunching parameter b in order to observe the changes of the quadrature points due to change of scale parameter a. We observed that if we increases the scale parameter constantly by 0.01 (1.876 → 1.896), each of the quadrature points will increase by a fixed value. For instance, the 10th quadrature point increases by 0.009334 when the scale parameter increases by 0.01.

Table 4.1 : 3 sets of 24 quadrature points for electron-Rb scattering at 50 eV with different scale parameter a and fixed bunching parameter b. The columns (setA 2 - setA 1) and (setA 3 - setA 2) is the values of difference of each quadrature point between different set of quadrature points.

	24 quadrature points (fixed $b = 0.063$)				
no.	setA 1 ($a = 1.876$)	setA 2 ($a = 1.886$)	setA 3 ($a = 1.896$)	setA 2 - setA 1	setA 3 - setA 2
1	0.017924	0.018019	0.018115	0.000096	0.000096
2	0.093740	0.094240	0.094739	0.000500	0.000500
3	0.227149	0.228360	0.229571	0.001211	0.001211
4	0.412489	0.414688	0.416887	0.002199	0.002199
5	0.640274	0.643687	0.647100	0.003413	0.003413
6	0.896007	0.900783	0.905559	0.004776	0.004776
7	1.159368	1.165548	1.171728	0.006180	0.006180
8	1.405238	1.412729	1.420220	0.007491	0.007491
9	1.608272	1.616845	1.625418	0.008573	0.008573
10	1.751014	1.760347	1.769681	0.009334	0.009334
11	1.831785	1.841549	1.851314	0.009764	0.009764
12	1.866821	1.876772	1.886723	0.009951	0.009951
13	1.885216	1.895265	1.905314	0.010049	0.010049
14	1.921079	1.931319	1.941559	0.010240	0.010240
15	2.008142	2.018847	2.029551	0.010704	0.010704
16	2.178883	2.190498	2.202112	0.011615	0.011614
17	2.467757	2.480912	2.494066	0.013154	0.013154
18	2.918517	2.934074	2.949631	0.015557	0.015557
19	3.598307	3.617488	3.636669	0.019181	0.019181
20	4.628169	4.652840	4.677510	0.024670	0.024670
21	6.261568	6.294945	6.328322	0.033377	0.033377
22	9.136608	9.185310	9.234013	0.048703	0.048703
23	15.412068	15.494222	15.576376	0.082154	0.082154
24	39.081922	39.290248	39.498573	0.208326	0.208325

Table 4.2 shows 3 sets of quadrature points for electron-Rb generated by varying the bunching parameter b with the fixed scale parameter a in order to observe the changes of the quadrature points due to bunching parameter b. We observe that as the bunching parameter b increases, the difference between the scale parameter a with each quadrature point increases (observing the columns (|1.886 - set 1|), (|1.886 - set 2|) and (|1.886 - set 3|) of 12th and 13th quadrature points). This roughly shows the idea of the role of bunching parameter b in clustering the points around the scale parameter a.

Table 4.2 : 3 sets of 24 quadrature points for electron-Rb scattering at 50 eV with different bunching parameters b and fixed scale parameter a. The columns (|1.886 - set 1|), (|1.886 - set 2|) and (|1.886 - set 3|) are the absolute difference between the scale parameter a with each quadrature points.

no	24 quadrature points (fixed a = 1.886)			|1.886 - set 1|	|1.886 - set 2|	|1.886 - set 3|
	set 1 (b = 0.053)	set 2 (b = 0.063)	set 3 (b = 0.073)			
1	0.018179	0.018019	0.017838	1.867821	1.867981	1.868162
2	0.095078	0.094240	0.093285	1.790922	1.791760	1.792715
3	0.230411	0.228360	0.226026	1.655589	1.657640	1.659974
4	0.418451	0.414688	0.410409	1.467549	1.471312	1.475591
5	0.649570	0.643687	0.637005	1.236430	1.242313	1.248995
6	0.908987	0.900783	0.891473	0.977013	0.985217	0.994527
7	1.175897	1.165548	1.153815	0.710103	0.720453	0.732185
8	1.424485	1.412729	1.399411	0.461515	0.473271	0.486589
9	1.628654	1.616845	1.603468	0.257346	0.269155	0.282532
10	1.770469	1.760347	1.748871	0.115531	0.125653	0.137129
11	1.848347	1.841549	1.833830	0.037653	0.044451	0.052170
12	1.879161	1.876772	1.874054	0.006839	0.009228	0.011946
	a = 1.886					
13	1.892860	1.895265	1.898008	0.006860	0.009265	0.012008
14	1.924274	1.931319	1.939370	0.038274	0.045319	0.053370
15	2.007584	2.018847	2.031742	0.121584	0.132847	0.145742
16	2.175469	2.190498	2.207735	0.289469	0.304498	0.321735
17	2.462434	2.480912	2.502137	0.576434	0.594912	0.616137
18	2.912207	2.934074	2.959224	1.026207	1.048074	1.073224
19	3.591888	3.617488	3.646966	1.705888	1.731488	1.760966
20	4.622485	4.652840	4.687828	2.736485	2.766840	2.801828
21	6.257491	6.294945	6.338157	4.371491	4.408945	4.452157
22	9.135225	9.185310	9.243142	7.249225	7.299310	7.357142
23	15.415410	15.494222	15.585280	13.529410	13.608222	13.699280
24	39.098630	39.290248	39.511721	37.212630	37.404248	37.625721

Table 4.3 : On-shell coordinate for electron-Rb scattering at 50 eV. $a = 1.886$ and $b = 0.063$.

channel	on-shell coordinate	weight
1 (5s)	1.9170037	-4.593757
2 (5p)	1.8864632	-0.450765
3 (4d)	1.8704297	-5.304390
4 (6s)	1.8685345	-3.353649
5 (6p)	1.8596407	1.996859
6 (5d)	1.8549138	4.909573
7 (7s)	1.8534129	6.109608
8 (7p)	1.8495792	10.706448

In principle, the quadrature mesh must cover the closely spaced points near the on-shell value of k. Table 4.3 shows the on-shell coordinates for each channel in the electron-Rb scattering at 50 eV. By following the previous observations, the value of scale parameter a and bunching parameter b can be adjusted until we obtain a satisfying set of quadrature points. Thus, we can avoid getting quadrature points which are too close to the on-shell values in order to avoid the singularity in the calculation due to the Green's function. The weights of the on-shell coordinates provide a glance on the quality of the quadrature points. A set of points is considered good when the weight for each channel is small. Table 4.1 is an example of reasonable low weights for each channel when $a = 1.886$ and $b = 0.063$. The weight of the on-shell value is merely a simple guide on getting a good calculation.

4.2.2 Convergence of the Cross Section

The convergence of the cross section is crucial in the calculations. In order for the cross sections to converge, the total angular momentum, *JMAX*, must be sufficiently large. Higher *JMAX* are needed as the incident energy increases.

Comparing the partial-wave (PW) sum and extrapolated PW sum in Table 4.4, we can observe a difference of 0.2% for the 5s channel and 53.2% for the 5p channel.

Table 4.4 : TCS for each channel in the unit of πa_0^2 obtained from the CC8 calculation at 100 eV. *JMAX* used for the calculation is 60. PW is the short form for partial-wave.

	5s	5p	4d	6s	6p	5d	7s	7p
PW sum	7.2419	16.942	0.61103	0.38637	0.56328	1.5515	0.48664	0.15836
integral	7.2433	24.226	0.74994	0.38422	0.59097	1.5525	0.48675	0.16073
extrapolated PW sum	7.2565	25.959	0.61419	0.39102	0.57183	1.5527	0.48979	0.1809

Table 4.5 : TCS for each channel in the unit of πa_0^2 obtained from the CC8 calculation for 100 eV. *JMAX* used for the calculation is 180.

	5s	5p	4d	6s	6p	5d	7s	7p
PW sum	10.328	26.355	0.64917	0.39588	0.58414	1.5558	0.4885	0.16963
integral	7.285	24.697	0.65669	0.38881	0.58183	1.5544	0.48866	0.16982
extrapolated PW sum	10.328	27.458	0.65322	0.39494	0.58443	1.5558	0.4885	0.16917

The large differences in 5p channel indicate that the cross sections of the 5p scattering are not converged yet. In order to get converged 5p cross sections, we need higher *JMAX* in the calculation. In Table 4.5, we increase the *JMAX* of the calculation to *JMAX* = 180 and we observe that there is a great improvement in the convergence of the 5p cross sections.
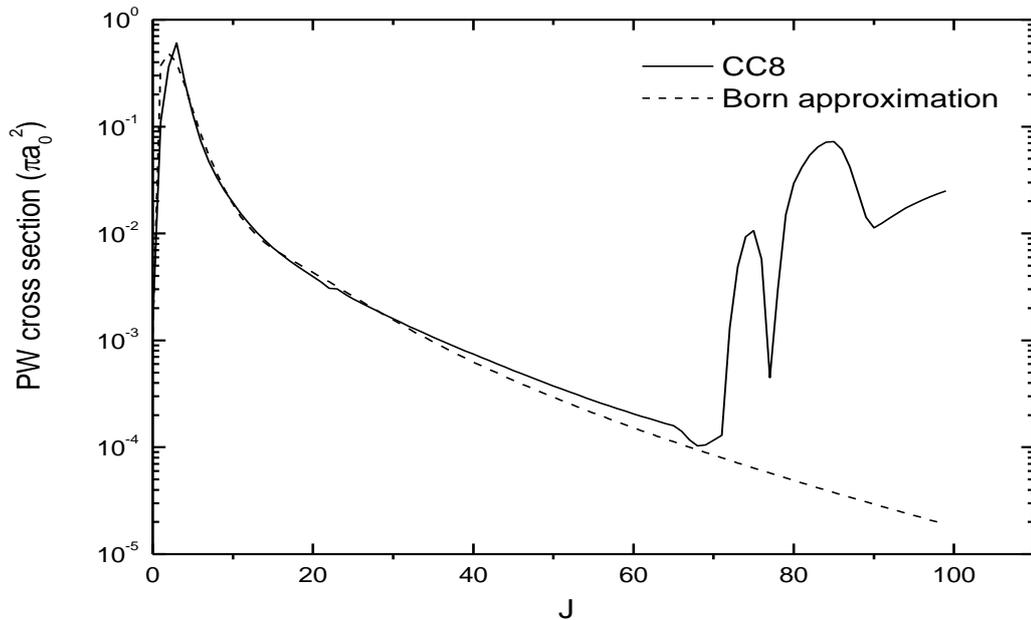
Fig. 4.1 : Partial-wave cross section for 5s channel in the unit of πa_0^2 obtained from the CC8 calculation for 100 eV. *JMAX*=100 and UBA is the uniterized Born approximation.

The differences of PW sum and extrapolated PW sum have dropped to 4.2% which is reasonably good. But, we observe another problem in Table 4.5. The PW sum and extrapolated PW sum from Table 4.5 differ a lot from Table 4.4. In order to trace down the problem, we investigate the PW cross section of the CC8 calculation.

We observe from Figure 4.1 that there are fluctuations in the CC8 PW cross section from J=60 onward due to the decrease in the quality of the calculation due to numerical inaccuracies and limitation of the computational resources, i.e. the CC8 calculation is inconsistent after $J = 60$. Since the Born Approximation PW cross section is good at $J = 60$ onward, we can get rid of the fluctuations problem by merging the calculations of CC and CCOM with the uniterized Born approximation (UBA). Let $JMAX1$ be the $JMAX$ of CC or CCOM before the quality of the calculations decreases and $JMAX2$ be the $JMAX$ of UBA. By using a merging program, we can merge the T-matrices of the CC/CCO and UBA calculations to produce a new T-matrix where the new $JMAX$ is the combination of $JMAX1$ and $JMAX2a$ ($JMAX1+1 \leq JMAX2a \leq JMAX2$). For example, at 100 eV, $JMAX = 180$ where $JMAX1 = 60$, JMAX2 = 180 and $JMAX2a$ ($JMAX1+1 \leq JMAX2a \leq JMAX2$) = 61-180.

Table 4.6 : TCS for each channel in the unit of πa_0^2. $JMAX$ used for the CC8 calculation is 180. The $JMAX$ used is the new $JMAX$ after the merging.

	5s	5p	4d	6s	6p	5d	7s	7p
PW sum	7.2531	25.354	0.7517	0.39361	0.59247	1.5526	0.48695	0.16162
integral	7.2636	24.181	0.75905	0.38699	0.59117	1.5528	0.48735	0.16162
extrapolated PW sum	7.2532	27.117	0.76037	0.39431	0.59467	1.5526	0.48695	0.16183

By using the merged T-matrices, the calculation yields converged cross sections for all the channels as shown in Table 4.6, as well as getting rid of the inconsistency in the CC8 calculation after J=60 as shown in Figure 4.2.
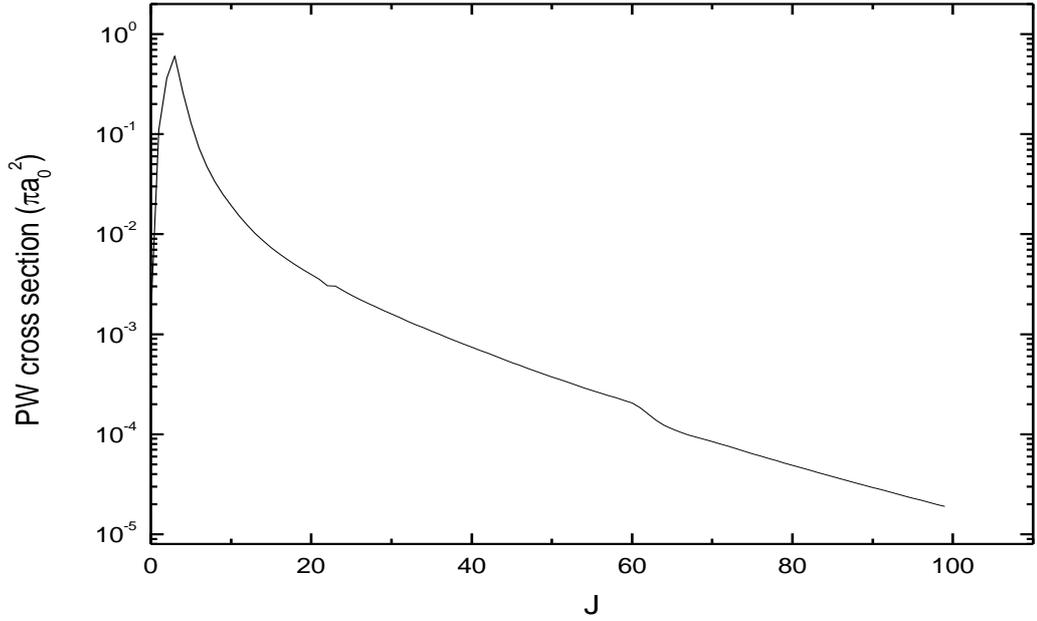
Fig. 4.2 : PW cross section for 5s channel in the unit of πa_0^2 for the CC8 calculation at 100 eV.

4.3 Numerical Details in Positron-Rb Scattering Calculations

4.3.1 Quadrature Points

Similar to the electron scattering case, the challenge once again lies in the distribution of the quadrature points. For the electron case, we adjust the values of scale and bunching parameters to distribute the quadrature points. For the positron case, since we have atomic on-shell coordinates as well as the Ps on-shell coordinates, so we need to distribute the quadrature points by using another method. The other bunching transformation method which is the 5-panels mesh is also known as Method B. Method B is explained in the work by Ratnavelu *et al.* (1996). In Method B, we distribute the quadrature points into 5 regions, determined by these parameters: smk1, smk2, smk3, dmk1, dmk2.

In the positron-Rb scattering, the difference between the on-shell values of the Ps and atomic states are quite large at intermediate and high energy (the difference
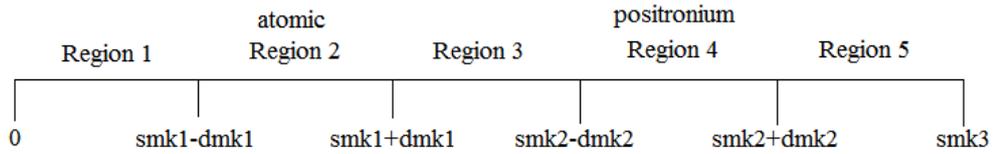
Fig. 4.3 : The quadrature points are distributed among the 5 regions as shown. The regions are determined by the values of smk1, smk2, smk3, dmk1 and dmk2.

increases when the energy increases). The quadrature points are distributed carefully among the 5 regions so that only the quadrature points in Region 2 will be close to the atomic channels on-shell coordinates while the quadrature points in Region 4 will be close to the Ps channels on-shell coordinates. The values of the parameters (smk1, smk2, smk3, dmk1, dmk2) need to be adjusted carefully to fulfil this condition. From Figure 4.3 we can see that smk1 and dmk1 will determine the "area" of Region 2 while smk2 and dmk2 determine the "area" of Region 4.

Table 4.7 - 4.11 show the effects of varying the parameters smk1, smk2, dmk1 and dmk2 on the distribution of the quadrature points in Region 2 and Region 4. Since we are using 68 quadrature points in the calculations, so Region 1 contains 1st to 24th points (24 points), Region 2 contains 25th to 36th points (12 points), Region 3 contains 37th to 46th points (10 points), Region 4 contains 47th to 58th points (12 points) and Region 5 contains 59th to 68th points (10 points).

By changing the magnitudes of smk1 and smk2, the values of the quadrature points will change according to the magnitudes of changes of smk1 and smk2. For example, by increasing the smk1 by 0.02 (Table 4.7 set C), all the quadrature points in Region 2 will be increased by 0.02 (Table 4.7 set C-set A). Similarly, by decreasing the smk2 by 0.03 (Table 4.8 set B), all the quadrature points in Region 4 will be decreased by 0.03 (Table 4.8 set B-set A).

Table 4.7 : The effects of varying smk1 on the distribution of quadrature points in Region 2. The smk1 of set B differs from set A by -0.03 and set C differs from set A by +0.02.

	smk1				
	set A	set B	set C	set B-set A	set C-set A
no	**1.1352**	**1.1052**	**1.1552**		
25	1.083925	1.053925	1.103925	-0.03	0.02
26	1.094336	1.064336	1.114336	-0.03	0.02
27	1.108743	1.078743	1.128743	-0.03	0.02
28	1.122034	1.092034	1.142034	-0.03	0.02
29	1.130672	1.100672	1.150672	-0.03	0.02
30	1.134381	1.104381	1.154381	-0.03	0.02
31	1.136019	1.106019	1.156019	-0.03	0.02
32	1.139728	1.109728	1.159728	-0.03	0.02
33	1.148366	1.118366	1.168366	-0.03	0.02
34	1.161658	1.131658	1.181658	-0.03	0.02
35	1.176064	1.146064	1.196064	-0.03	0.02
36	1.186475	1.156475	1.206475	-0.03	0.02

Table 4.8 : The effects of varying smk2 on the distribution of quadrature points in Region 4. The smk2 of set B differs from set A by -0.03 and set C differs from set A by +0.02.

	smk2				
	set A	set B	set C	set B-set A	set C-set A
no	**1.6724**	**1.6424**	**1.6924**		
47	1.54991	1.51991	1.56991	-0.03	0.02
48	1.574781	1.544781	1.594781	-0.03	0.02
49	1.609196	1.579196	1.629196	-0.03	0.02
50	1.640948	1.610948	1.660948	-0.03	0.02
51	1.661584	1.631584	1.681584	-0.03	0.02
52	1.670443	1.640443	1.690443	-0.03	0.02
53	1.674357	1.644357	1.694357	-0.03	0.02
54	1.683216	1.653216	1.703216	-0.03	0.02
55	1.703852	1.673852	1.723852	-0.03	0.02
56	1.735604	1.705604	1.755604	-0.03	0.02
57	1.770019	1.740019	1.790019	-0.03	0.02
58	1.79489	1.76489	1.81489	-0.03	0.02

The dmk1 and dmk2 change the quadrature points by a fixed "rate". In Table 4.9 and 4.10, the values of dmk1 and dmk2 of set B differ from set A by -0.001 and set C by +0.002. If we take the ratio of the difference, it will be 1:2. It is not difficult for us to notice that the ratio of |set B-set A| and |set C-set A| is also about 1:2. So, dmk1 and

dmk2 directly control the ranges of Region 2 and Region 4. By increasing or decreasing

dmk1 and dmk2, we can increase or decrease the ranges of Region 2 and Region 4

accordingly.

Table 4.9 : The effects of varying dmk1 on the distribution of quadrature points in Region 2. The dmk1 of set B differs from set A by -0.001 and set C differs from set A by +0.002.

	dmk1				
	set A	set B	set C	|set B-set A|	|set C-set A|
no	0.054	0.053	0.056		
25	1.083925	1.084875	1.082026	0.0009496	0.001899
26	1.094336	1.095093	1.092823	0.0007568	0.0015134
27	1.108743	1.109232	1.107763	0.0004899	0.0009799
28	1.122034	1.122278	1.121547	0.0002438	0.0004877
29	1.130672	1.130756	1.130505	8.39E-05	0.0001677
30	1.134381	1.134396	1.13435	1.52E-05	3.03E-05
31	1.136019	1.136004	1.13605	1.52E-05	3.03E-05
32	1.139728	1.139644	1.139895	8.39E-05	0.0001677
33	1.148366	1.148122	1.148854	0.0002438	0.0004877
34	1.161658	1.161168	1.162637	0.0004899	0.0009799
35	1.176064	1.175307	1.177577	0.0007568	0.0015134
36	1.186475	1.185526	1.188374	0.0009496	0.001899

Table 4.10 : The effects of varying dmk2 on the distribution of quadrature points in Region 4. The dmk2 of set B differs from set A by -0.001 and set C differs form set A by +0.002.

	dmk2				
	set A	set B	set C	|set B-set A|	|set C-set A|
no	0.129	0.128	0.131		
47	1.54991	1.550859	1.548011	0.0009495	0.0018991
48	1.574781	1.575538	1.573268	0.0007567	0.0015135
49	1.609196	1.609686	1.608216	0.0004899	0.0009799
50	1.640948	1.641192	1.640461	0.0002438	0.0004876
51	1.661584	1.661668	1.661416	8.38E-05	0.0001677
52	1.670443	1.670458	1.670413	1.51E-05	3.04E-05
53	1.674357	1.674342	1.674388	1.51E-05	3.04E-05
54	1.683216	1.683132	1.683384	8.38E-05	0.0001677
55	1.703852	1.703608	1.704339	0.0002438	0.0004876
56	1.735604	1.735114	1.736584	0.0004899	0.0009799
57	1.770019	1.769262	1.771533	0.0007567	0.0015135
58	1.79489	1.793941	1.79679	0.0009495	0.0018991

In summary, smk1 and smk2 in Method B are similar to the scale parameter a in Method A while dmk1 and dmk2 in Method B are similar to the bunching parameter b in Method A.

By adjusting the values of smk1, smk2, dmk1 and dmk2 carefully, we want to distribute the quadrature points in Region 2 and Region 4 so that the points in the regions can cover the closely spaced points near the on-shell coordinates of the atomic and the Ps channels. Similarly to the electron-case, a set of points is considered to be good when the weight of each channel is small.

At lower energy ($\leq$6 eV), the on-shell values of the atomic and Ps states will overlap, thus Method B will be less effective in distributing the quadrature points.

Method B is applicable when the atomic and Ps on-shell coordinates do not overlap (as shown in Figure 4.4). In Method B, we can distribute the quadrature points in such a way that the quadrature points in Region 2 will only associate with the atomic channels while the quadrature points in Region 4 will associate with the Ps channels.

Table 4.11 : The on-shell coordinates for the CC(5,3) calculation at 6 eV and 15 eV.

	On-shell coordinates	
States	**6 eV**	**15 eV**
Ps(1s)	0.989718	1.565089
Rb(5s)	0.544767	1.015752
Rb(5p)	0.424983	0.956865
Rb(4d)	0.34695	0.924854
Ps(2s)	0.479104	1.30365
Ps(2p)	0.479104	1.30365
Rb(6s)	0.336584	0.921015
Rb(6p)	0.283073	0.902835
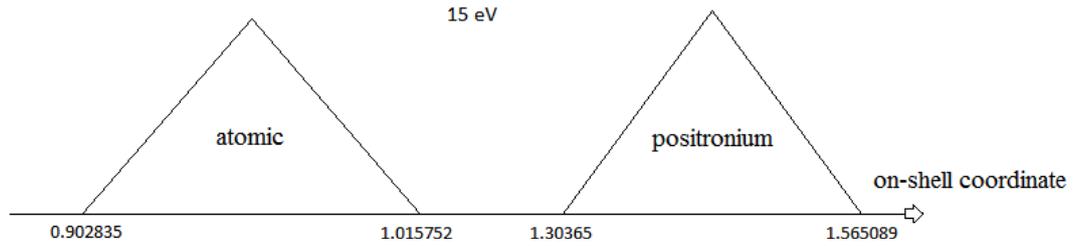
Fig. 4.4 : The illustration of the on-shell coordinates for CC(5,3) calculations at 15 eV.
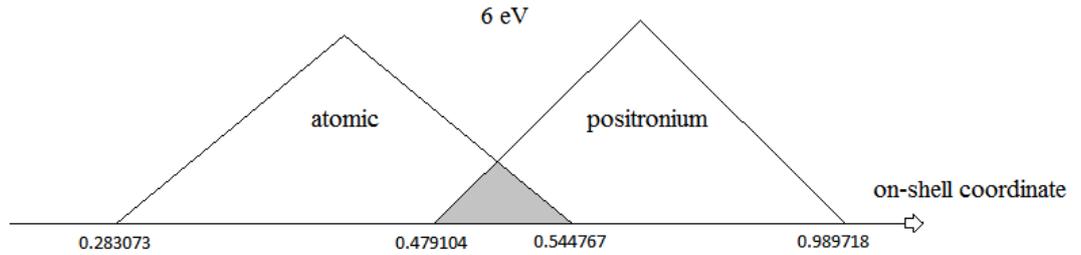
Fig. 4.5 : The illustration of the on-shell coordinates for CC(5,3) calculations at 6 eV. The grey area is the overlapping region of atomic and Ps on-shell coordinates.

When the atomic on-shell coordinates (Region 2) are overlaping with Ps on-shell coordinates (Region 5) as shown in Figure 4.5, Region 3 vanishes and Method B is not suitable anymore because Method B is a 5 regions based method. We might face problems in distributing the quadrature points and the calculations will be inaccurate as the quadrature mesh does not cover points near the on-shell coordinates effectively. The best solution for this problem is to resort to Method A for those lower energies for which their on-shell coordinates overlap.

4.4   Numerical Convergence of the Lippmann-Schwinger Solution

In this section, the numerical convergence of the LS solution is studied. The CC(5,3) and CC(8,6) models are used for this purpose. The CC(5,3) calculation consists of 5 Rb atomic states and 3 Ps states while the CC(8,6) consists of 8 Rb atomic states and 6 Ps states. For testing purposes, we only choose certain energies to represent certain scattering energy range. For instance, we choose 5 eV and 10 eV to represent low scattering energies, and 20 eV as the intermediate scattering energy.

As discussed in Sections 4.1 and 4.2, we have applied 2 types of quadrature point meshes known as Method A and Method B in our calculations. Method B consists of 5 regions, thus we labelled the integration mesh as $R(m_1, m_2, m_3, m_4, m_5)$. The term m_i indicates that the number of quadrature points distributed in the ith ($i = 1, 2, 3, 4, 5$) region. Each region can contain 6-48 quadrature points and the total quadrature points (Q-points) in all the regions can reach a maximum of 116 points.

The numerical convergence test of Method A is quite straight forward. We used a variety of total quadrature points such as 24, 32 and 48 Q-points in the electron case, and 32, 48, 56, 64 and 84 Q-points in the positron case. For Method B, we used various combinations of quadrature distribution in all the regions. The combinations are obtained by changing the quadrature points in each region. First of all, we randomly define an integration mesh. Then, we increase or decrease the quadrature points in a particular region but remain the number of quadrature points in the other regions. By doing this, we can observe the changes in the cross section of each channel by the change of the integration mesh.

Our main goal for the numerical convergence test is to obtain an optimal mesh that yields the most converged cross sections in all the channels. Besides, we also want to choose the mesh that consumes less computational time but still is able to produce converged results. Generally, the lesser the total quadrature points, the lesser the computational time required to perform the calculations.

Table 4.12 - 4.14 show the numerical convergence of CC(5,3) calculation at 5 eV, 10 eV and 20 eV. We used Method A for 5 eV as discussed in Section 4.2. The calculations are done using $JMAX = 10$ as it is sufficient for the testing purpose. At 5 eV (Table 4.12), we start by choosing 32 Q-points, followed by 48, 56, 64 and 84 points. We choose 48 Q-points as it yields converged cross sections in almost all the channels,

which are less than 3% different compared to the higher Q-points calculations. Although the 48 Q-points mesh does not yield good convergence in Rb(6s) and Rb(6p) channels (with more than 7% absolute error), the contributions from both of these channels are relatively insignificant (less than 1% of the TCS), so the errors can be ignored. Although 32 Q-points mesh can also yield converged results and consume lesser computational time, we will still stick to 48 Q-points. The only reason we choose 48 Q-points rather than 32 Q-points is because we can obtain low weight in each channel by using 48 Q-points compared to 32 Q-points (refer to Section 4.1.1 for the explanation of the role of weight in each channel).

Table 4.13 is the tabulation of numerical convergence of CC(5,3) calculation at 10 eV using JMAX = 10. Our first randomly chosen integration mesh is R(24,12,10,12,10). We compared this mesh with R(24,12,6,12,10) and we concluded that the number of Q-points in Region 3 does not affect the result of the calculation very much (less than 0.1% of absolute errors in all channels). We continue our analysis by comparing R(24,12,10,12,10) and R(24,12,10,12,6) and we observe that the number of Q-points in Region 5 also does not affect the result of the calculation very much (<1% of absolute error). After dealing with Region 3 and Region 5, we wanted to deal with Region 1. We analyse the results which are obtained by using R(20,12,10,12,10), R(24,12,10,12,10), R(20,16,10,12,10), R(24,16,10,12,10), R(20,12,10,16,10) and R(24,12,10,16,10). The results are compared in this way:

$$R(20,12,10,12,10) \longleftrightarrow R(24,12,10,12,10);$$
$$R(20,16,10,12,10) \longleftrightarrow R(24,16,10,12,10);$$
$$R(20,16,10,12,10) \longleftrightarrow R(24,16,10,12,10);$$

We can observe that by increasing the number of Q-points in Region 1 from 20 points to 24 points, the absolute errors in all the channels are less than 1%, which is very low and thus can be ignored. We choose to allocate 24 Q-points in this region. We

could have allocated 20 points in Region 1 to decrease the computational time but we opt to sacrifice a bit of the computational time for better converged results.

Region 2 and Region 4 consist of the Q-points which are needed to cover the on-shell coordinates of the atomic and Ps channels. So, the number of Q-points must be large enough to cover all the coordinates but low enough to save computational time. We analyse the results of R(24,12,10,12,10), R(24,16,10,12,10), R(24,12,10,16,10), R(24,16,10,16,10), R(24,20,10,16,10), R(24,16,10,20,10) and R(24,20,10,20,10). R(24,12,10,12,10) yields sufficiently converged cross sections in all the channels by having less than 2% absolute errors compared to all the other meshes. Considering the satisfying converged results and shorter computational time, we concluded that R(24,12,10,12,10) is the optimal mesh for 10 eV.

We carried out similar numerical convergence test on the CC(5,3) calculations at 20 eV (Table 4.14) and we still observe that R(24,12,10,12,10) is the best mesh at 20 eV with the absolute error less than 0.7%. As a conclusion, we decided to do all the CC(5,3) calculations by using this mesh.

We also undertook numerical convergence tests on the CC(8,6) calculations at 5 eV, 10 eV and 20 eV using $JMAX = 10$ (Table 4.15 - 4.17). The analysis procedures are identical to the analysis of the CC(5,3) calculations. We faced challenges in choosing the optimal mesh for CC(8,6) at 5 eV as the cross sections for all the channels have barely converged.

Thus, we decided to analyse some of the major channels (that is, Rb(5s,5p,6s,6p) and Ps(1s,2s,2p)). We found that the TCS calculated using 32 Q-points (about $180\pi a_0^2$) does not converge with the other meshes (about $176\pi a_0^2$), so it is not the optimal mesh. Then, we studied the convergence using 48 Q-points. The 48 Q-points mesh produces satisfactorily converged cross sections in almost all the major channels with the highest

absolute error to be less than 12.5%. Thus, we choose the 48 Q-points mesh as the optimal mesh for CC(8,6) at 5 eV because this mesh can yield converged results at lowest computational resources.

The numerical analysis on the results of the CC(8,6) calculations at 10 eV and 20 eV is similar to the CC(5,3) case and thus we will not repeat the explanation. We conclude that the integration mesh, R(24,12,10,12,10), yielded the most converged results in all the testing energies by having the average absolute error to be <5% and <4% in most all of the channels at 10 eV and 20 eV, respectively.

In comparison to the CC(5,3), it is harder to decide the optimal mesh for CC(8,6) because we have more channels to be considered in CC(8,6). Note that most of the high magnitude errors (absolute error >5%) come from the Rb(6s,6p,7s,7p) and Ps(3s,3p,3d) channels. Since these channels are not the dominant channels in the positron-Rb scattering system, they are less significant in the analysis. We finally decide to use the R(24,12,10,12,10) mesh in all the CC(8,6) calculations.

No numerical convergence test on the CC(5,6) and CC(8,3) calculations was undertaken as we expect the results to be converged. This is due to the fact the CC(5,3) and CC(8,6) calculations have the smallest and largest atomic and Ps states among our present calculations. If the CC(5,3) and CC(8,6) calculations can produce converged results, we expect the same on the CC(5,6) and CC(8,3) calculations.

Table 4.12a : The numerical convergence for CC(5,3) calculation at 5 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Cross Section (πa_0^2)								
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	TCS
	32	9.40	76.65	57.20	24.97	7.45	9.10	1.44	1.27	180.42
	48	9.34	76.21	57.93	23.96	7.40	9.17	1.46	1.18	179.65
Method A	56	9.11	76.94	58.57	24.27	7.30	9.10	1.35	1.14	180.81
	64	9.41	76.67	58.26	23.55	7.45	9.20	1.35	1.15	180.05
	84	9.21	76.80	58.34	23.94	7.34	9.14	1.34	1.15	180.29

Table 4.12b : The absolute errors between 48 Q-points mesh with the other meshes for CC(5,3) calculation at 5 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Absolute Errors (%)								
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	TCS
	32	0.65	0.58	1.26	4.20	0.71	0.86	1.65	7.46	0.43
	48									
Method A	56	2.47	0.96	1.10	1.29	1.43	0.81	7.46	3.89	0.65
	64	0.83	0.60	0.57	1.72	0.71	0.27	7.35	2.74	0.22
	84	1.40	0.78	0.71	0.08	0.88	0.42	8.11	3.15	0.36

Table 4.13a : The numerical convergence for CC(5,3) calculation at 10 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Cross Section (πa_0^2)								
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	TCS
R(20,12,10,12,10)	64	2.01	43.30	165.57	39.45	5.69	18.24	0.92	3.11	123.33
R(24,12,6,12,10)	64	2.03	43.50	165.68	39.53	5.66	18.20	0.93	3.14	123.41
R(24,12,10,12,6)	64	2.05	43.52	165.58	39.57	5.67	18.23	0.93	3.13	123.47
R(24,12,10,12,10)	68	2.03	43.50	165.66	39.53	5.66	18.20	0.93	3.14	123.41
R(20,16,10,12,10)	68	2.00	43.24	164.43	39.51	5.66	18.17	0.94	3.10	123.15
R(20,12,10,16,10)	68	2.01	43.30	166.06	39.51	5.77	18.14	0.93	3.11	123.29
R(24,16,10,12,10)	72	2.02	43.45	164.42	39.59	5.63	18.12	0.95	3.13	123.24
R(24,12,10,16,10)	72	2.03	43.49	166.15	39.60	5.74	18.10	0.93	3.14	123.37
R(24,16,10,16,10)	76	2.02	43.44	164.90	39.66	5.71	18.03	0.95	3.13	123.19
R(24,20,10,16,10)	80	2.02	43.39	164.42	39.67	5.70	18.01	0.94	3.14	123.13
R(24,16,10,20,10)	80	2.02	43.43	164.89	39.70	5.76	17.97	0.95	3.13	123.18
R(24,20,10,20,10)	84	2.02	43.38	164.41	39.71	5.74	17.95	0.94	3.14	123.11

Table 4.13b : The absolute errors between R(24,12,10,12,10) mesh with the other meshes for CC(5,3) calculation at 10 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Absolute Errors (%)								
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	TCS
R(20,12,10,12,10)	64	0.92	0.44	0.05	0.20	0.43	0.20	0.87	0.88	0.06
R(24,12,6,12,10)	64	0.00	0.00	0.01	0.01	0.01	0.01	0.03	0.01	0.00
R(24,12,10,12,6)	64	0.92	0.06	0.05	0.11	0.17	0.13	0.56	0.25	0.05
R(24,12,10,12,10)	68	■	■	■	■	■	■	■	■	■
R(20,16,10,12,10)	68	1.30	0.59	0.74	0.06	0.13	0.20	0.68	1.27	0.21
R(20,12,10,16,10)	68	0.84	0.46	0.24	0.04	1.81	0.34	0.73	0.84	0.10
R(24,16,10,12,10)	72	0.50	0.12	0.75	0.15	0.63	0.43	1.58	0.32	0.14
R(24,12,10,16,10)	72	0.08	0.01	0.30	0.17	1.37	0.54	0.15	0.04	0.03
R(24,16,10,16,10)	76	0.41	0.13	0.46	0.33	0.73	0.96	1.74	0.28	0.18
R(24,20,10,16,10)	80	0.44	0.25	0.75	0.35	0.55	1.05	1.24	0.03	0.23
R(24,16,10,20,10)	80	0.41	0.14	0.46	0.44	1.59	1.30	1.77	0.28	0.19
R(24,20,10,20,10)	84	0.43	0.26	0.75	0.47	1.40	1.40	1.27	0.02	0.24

Table 4.14a : The numerical convergence for CC(5,3) calculation at 20 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Cross Section (πa_0^2)								
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	TCS
R(24,12,10,12,10)	68	0.5589	16.3650	25.7320	6.7725	0.5429	1.1988	0.5988	1.4377	53.21
R(24,16,10,12,10)	72	0.5588	16.3650	25.7250	6.7669	0.5432	1.1985	0.6011	1.4382	53.20
R(24,12,10,16,10)	72	0.5592	16.3660	25.7320	6.7724	0.5421	1.1989	0.5987	1.4375	53.21
R(24,16,10,16,10)	76	0.5590	16.3660	25.7250	6.7667	0.5424	1.1986	0.6011	1.4380	53.20
R(24,20,10,16,10)	80	0.5584	16.3650	25.7300	6.7688	0.5418	1.1982	0.6007	1.4363	53.20
R(24,16,10,20,10)	80	0.5591	16.3660	25.7250	6.7667	0.5422	1.1987	0.6010	1.4380	53.20
R(24,20,10,20,10)	84	0.5585	16.3660	25.7300	6.7685	0.5407	1.1986	0.6006	1.4361	53.20

Table 4.14b : The absolute errors between R(24,12,10,12,10) mesh with the other meshes for CC(5,3) calculation at 20 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Absolute Errors (%)								
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	TCS
R(24,12,10,12,10)	68									
R(24,16,10,12,10)	72	0.03	0.00	0.03	0.08	0.05	0.03	0.39	0.03	0.02
R(24,12,10,16,10)	72	0.04	0.01	0.00	0.00	0.14	0.01	0.01	0.01	0.00
R(24,16,10,16,10)	76	0.01	0.01	0.03	0.09	0.09	0.02	0.38	0.02	0.02
R(24,20,10,16,10)	80	0.10	0.00	0.01	0.05	0.20	0.05	0.32	0.10	0.01
R(24,16,10,20,10)	80	0.03	0.01	0.03	0.09	0.13	0.01	0.38	0.02	0.02
R(24,20,10,20,10)	84	0.08	0.01	0.01	0.06	0.39	0.02	0.31	0.11	0.01

Table 4.15a : The numerical convergence for CC(8,6) calculation at 5 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Cross Section (πa_0^2)														
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	Rb5d	Rb7s	Ps3s	Ps3p	Ps3d	Rb7p	TCS
Method A	32	5.90	77.25	53.52	22.87	4.63	6.50	2.10	1.92	1.44	0.48	0.54	1.52	1.59	0.73	180.98
	48	5.41	74.29	53.92	21.51	4.29	6.73	2.19	2.01	2.10	0.72	0.28	1.06	1.53	0.46	176.50
	56	5.65	75.14	50.94	22.63	4.39	6.89	1.92	1.77	2.10	0.56	0.37	1.53	1.61	0.67	176.17
	64	5.58	74.36	51.01	23.71	4.81	7.14	1.71	1.90	1.87	0.66	0.51	1.53	1.63	0.46	176.88
	84	5.68	73.83	51.16	22.64	4.24	6.97	1.92	1.81	1.94	0.55	0.57	1.70	1.71	0.54	175.25

Table 4.15b : The absolute errors between 48 Q-points mesh with the other meshes for CC(8,6) calculation at 5 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Absolute Errors (%)														
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	Rb5d	Rb7s	Ps3s	Ps3p	Ps3d	Rb7p	TCS
Method A	32	9.09	3.99	0.74	6.33	7.87	3.40	4.22	4.78	31.57	33.02	93.11	43.53	3.98	56.02	2.54
	48															
	56	4.46	1.14	5.52	5.20	2.18	2.33	12.06	11.93	0.15	21.22	32.02	44.78	5.38	44.52	0.19
	64	3.19	0.10	5.38	10.23	12.09	5.97	21.93	5.77	10.88	7.89	80.62	44.93	6.58	0.77	0.22
	84	5.06	0.62	5.11	5.24	1.19	3.53	12.15	10.32	7.84	22.94	102.68	60.66	11.79	15.53	0.71

Table 4.16a : The numerical convergence for CC(8,6) calculation at 10 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Cross Section (πa_0^2)														
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	Rb5d	Rb7s	Ps3s	Ps3p	Ps3d	Rb7p	TCS
R(20,12,10,12,10)	64	1.27	36.99	33.22	20.18	4.21	8.44	1.21	2.12	3.41	0.55	1.63	4.30	4.80	0.92	123.24
R(24,12,6,12,10)	64	1.32	37.57	32.96	20.41	4.04	8.44	1.10	2.17	3.33	0.60	1.68	4.61	4.46	0.94	123.62
R(24,12,10,12,6)	64	1.25	37.81	32.89	20.39	4.03	8.43	1.10	2.16	3.33	0.59	1.68	4.65	4.50	0.95	123.76
R(24,12,10,12,10)	68	1.32	37.57	32.96	20.41	4.04	8.44	1.10	2.17	3.33	0.60	1.68	4.61	4.46	0.94	123.62
R(20,16,10,12,10)	68	1.26	37.02	33.06	20.15	4.20	8.45	1.16	2.23	3.35	0.54	1.61	4.33	4.80	1.02	123.16
R(20,12,10,16,10)	68	1.28	36.98	33.44	20.21	4.27	8.43	1.22	2.15	3.37	0.55	1.60	3.96	4.85	0.94	123.27
R(24,16,10,12,10)	72	1.31	37.60	32.78	20.38	4.04	8.46	1.05	2.30	3.27	0.58	1.66	4.66	4.45	1.03	123.55
R(24,12,10,16,10)	72	1.33	37.53	33.15	20.47	4.13	8.40	1.10	2.19	3.31	0.60	1.58	4.32	4.51	0.95	123.57
R(24,16,10,16,10)	76	1.32	37.55	32.97	20.44	4.13	8.42	1.05	2.32	3.26	0.58	1.57	4.36	4.50	1.03	123.50
R(24,20,10,16,10)	80	1.32	37.53	33.04	20.51	4.13	8.47	1.08	2.25	3.27	0.57	1.59	4.39	4.49	1.01	123.63
R(24,16,10,20,10)	80	1.31	37.61	33.02	20.44	4.12	8.42	1.05	2.33	3.27	0.58	1.55	4.41	4.58	1.05	123.73
R(24,20,10,20,10)	84	1.31	37.59	33.09	20.51	4.12	8.47	1.08	2.25	3.28	0.57	1.57	4.43	4.57	1.02	123.86

Table 4.16b : The absolute errors between R(24,12,10,12,10) mesh with the other meshes for CC(8,6) calculation at 10 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Absolute Errors (%)														
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	Rb5d	Rb7s	Ps3s	Ps3p	Ps3d	Rb7p	TCS
R(20,12,10,12,10)	64	4.33	1.54	0.81	1.15	4.13	0.01	9.33	2.21	2.26	8.23	2.88	6.53	7.61	1.64	0.31
R(24,12,6,12,10)	64	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00
R(24,12,10,12,6)	64	5.75	0.64	0.20	0.08	0.34	0.19	0.04	0.16	0.15	0.70	0.29	0.91	0.79	1.77	0.11
R(24,12,10,12,10)	68															
R(20,16,10,12,10)	68	5.17	1.48	0.32	1.28	3.91	0.05	5.14	2.72	0.52	9.21	3.87	5.93	7.44	8.53	0.37
R(20,12,10,16,10)	68	3.57	1.56	1.48	0.98	5.75	0.09	10.53	0.64	1.29	7.23	4.69	14.03	8.74	0.27	0.28
R(24,16,10,12,10)	72	1.13	0.06	0.54	0.15	0.08	0.16	4.91	6.20	1.77	2.46	0.88	1.21	0.38	9.29	0.06
R(24,12,10,16,10)	72	0.35	0.11	0.59	0.28	2.38	0.51	0.13	1.00	0.49	0.64	5.54	6.28	1.02	0.78	0.04
R(24,16,10,16,10)	76	0.63	0.06	0.04	0.12	2.30	0.27	4.80	7.19	2.16	1.93	6.50	5.24	0.75	10.05	0.10
R(24,20,10,16,10)	80	0.28	0.12	0.27	0.47	2.30	0.33	2.08	3.64	1.96	4.32	5.22	4.75	0.58	7.11	0.01
R(24,16,10,20,10)	80	1.13	0.11	0.19	0.16	1.92	0.29	4.96	7.43	1.80	2.38	7.75	4.22	2.51	11.61	0.09
R(24,20,10,20,10)	84	0.76	0.05	0.42	0.51	1.91	0.31	2.30	3.84	1.59	4.82	6.28	3.76	2.32	8.72	0.19

Table 4.17a : The numerical convergence for CC(8,6) calculation at 20 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Cross Section														
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	Rb5d	Rb7s	Ps3s	Ps3p	Ps3d	Rb7p	TCS
R(24,12,10,12,10)	68	0.565	16.349	25.046	6.337	0.112	0.564	0.623	1.163	0.705	0.061	0.394	0.948	0.242	0.301	53.41
R(24,16,10,12,10)	72	0.572	16.431	24.979	6.328	0.117	0.555	0.626	1.154	0.706	0.059	0.379	0.937	0.237	0.306	53.38
R(24,12,10,16,10)	72	0.572	16.353	25.043	6.339	0.113	0.562	0.622	1.164	0.706	0.061	0.385	0.983	0.226	0.302	53.43
R(24,16,10,16,10)	76	0.579	16.435	24.976	6.331	0.118	0.552	0.625	1.154	0.706	0.059	0.371	0.970	0.221	0.307	53.40
R(24,20,10,16,10)	80	0.567	16.361	25.059	6.342	0.111	0.562	0.621	1.176	0.694	0.060	0.386	0.983	0.227	0.314	53.46
R(24,16,10,20,10)	80	0.575	16.438	24.979	6.325	0.119	0.553	0.625	1.154	0.705	0.058	0.370	0.959	0.223	0.306	53.39
R(24,20,10,20,10)	84	0.564	16.365	25.062	6.337	0.112	0.563	0.621	1.175	0.693	0.060	0.386	0.971	0.229	0.312	53.45

Table 4.17b : The absolute errors between R(24,12,10,12,10) mesh with the other meshes for CC(8,6) calculation at 20 eV (*JMAX* = 10).

Integrating mesh	Total Q-Points	Absolute Errors (%)														
		Ps1s	Rb5s	Rb5p	Rb4d	Ps2s	Ps2p	Rb6s	Rb6p	Rb5d	Rb7s	Ps3s	Ps3p	Ps3d	Rb7p	TCS
R(24,12,10,12,10)	68															
R(24,16,10,12,10)	72	1.21	0.50	0.27	0.13	4.02	1.64	0.47	0.82	0.08	3.55	3.98	1.11	2.29	1.63	0.05
R(24,12,10,16,10)	72	1.23	0.02	0.01	0.03	0.65	0.46	0.13	0.07	0.06	0.08	2.28	3.73	6.82	0.30	0.04
R(24,16,10,16,10)	76	2.57	0.53	0.28	0.09	4.89	2.17	0.31	0.77	0.14	3.65	5.89	2.38	8.93	1.90	0.01
R(24,20,10,16,10)	80	0.42	0.07	0.05	0.09	1.00	0.39	0.31	1.11	1.56	0.52	2.03	3.78	6.44	4.07	0.10
R(24,16,10,20,10)	80	1.85	0.54	0.27	0.18	5.62	2.02	0.36	0.83	0.03	3.74	6.24	1.19	7.86	1.56	0.04
R(24,20,10,20,10)	84	0.13	0.10	0.06	0.00	0.25	0.24	0.30	1.03	1.73	0.65	2.21	2.51	5.33	3.69	0.08