

1 RAS-CI Contraction scheme ($N_\alpha = N_\beta$)

Scheme for the em RAS_Contrac(V,ItrRAS,M,N) routine ($N_{alpha} = N_{beta}$):

```
1 use Reduced_Lists
2 use Addressing
3 read Fock, integrals
4 set parameters & dimensions
5 allocate amplitude vectors (jB) & responses (jR)
6 do Roots
7   read jB
8   do i1 = 1,M
9
10    if [Hole .and. N.le.(M-1)]
11      do MNAHa = MIN: LMIN
12         $F_{is} :< Act|F|Hole >$ 
13        build: LAHa(MNAHa), SgnAHa, doLAHa=true
14    enddo; endif
15    if [Part]
16      do MNAPa = MIN1: LMIN1
17         $F_{as} :< Act|F|Part >$ 
18        build: LAPa(MNAPa), SgnAPa, doLAPa=true
19    enddo; endif
20
21    do i2 = 1,i1
22      define: IFock, iXvv, iXoo1, iXoo2
23
24      if [Hole .and. N.le.(M-1)]
25        RAS_FormXah: LAHb(MNAHb), SgnAHb, doLAHb=true
26        do LAHa
27          do LAHb
28             $< HoleA|V|HoleB >$  (iXoo1)
29            if [i1.NE.i2]  $< HoleA|V|HoleB >$  (iXoo2)
30          enddo; enddo; endif
31        if [Part]
32          RAS_FormXap: LAPb(MNAPb), SgnAPb, doLAPb=true
33          do MNAPa: LAPa
34            do MNAPb: LAPb
35               $< PartA|V|PartB >$  (iXvv)
36              if [i1.NE.i2]  $< PartA|V|PartB >$  (iXvv)
37            enddo
38            if [doLAHb]
39              do MNAHb: LAHb
40                 $< PartA|V|HoleB >$ 
41            enddo; endif; enddo; endif
42
43          if [i1.NE.i2]
44
45            if [doLAHa .and. doLAPb]
46              do MNAHa: LAHa
47                do MNAPb: LAPb
48                   $(sp|s'h) = < Part|V|Act > < Act|V|Hole >$ 
49            enddo; enddo; endif
50
51          if [(M-2).ge.(N-1)]
```

```

52     do M2N1: LM2N1
53          $F_{ss'} = \langle Act|F|Act \rangle \langle Act|Act \rangle$ 
54         if [Hole]
55              $F_{ss'} = \langle Act|F|Act \rangle \langle Hole|Hole \rangle$ 
56              $(ss'|ij) = \langle Hole|V|Hole \rangle \langle Act|V|Act \rangle$ 
57         endif
58         if [Part]
59              $(ss'|ab) = \langle Part|V|Part \rangle \langle Act|V|Act \rangle$ 
60              $F_{ss'} = \langle Act|F|Act \rangle \langle Part|Part \rangle$ 
61             if [Hole]
62                  $(ps|hs') = \langle Hole|V|Part \rangle \langle Act|Act \rangle$ 
63         endif; endif; enddo; endif
64
65     if [Hole .and. (M-2).ge.N]
66         do M2N: LM2N
67              $F_{ss'} = \langle Hole|F|Hole \rangle \langle Act|Act \rangle$ 
68              $(ss'|ij) = \langle Hole|V|Hole \rangle \langle Act|Act \rangle$ 
69              $(si|s'j) = \langle Hole|V|Hole \rangle \langle Act|Act \rangle$ 
70         enddo; endif
71     if [Part .and. N.ge.2]
72         do M2N2: LM2N2
73              $(ss'|ab) = \langle Part|V|Part \rangle \langle Act|Act \rangle$ 
74              $(sa|s'b) = \langle Part|V|Part \rangle \langle Act|Act \rangle$ 
75              $F_{ss'} = \langle Part|F|Part \rangle \langle Act|Act \rangle$ 
76         enddo; endif
77
78     else ! [i1.EQ.i2]
79
80         do M1N1: LM1N1
81              $F_{ss} = \langle Act|F|Act \rangle \langle Act|Act \rangle$ 
82             if [Hole]
83                  $F_{ss} = \langle Act|F|Act \rangle \langle Hole|Hole \rangle$ 
84                  $(ss|ij) = \langle Hole|V|Hole \rangle \langle Act|V|Act \rangle$ 
85             endif
86             if [Part]
87                  $(ss|ab) = \langle Part|V|Part \rangle \langle Act|V|Act \rangle$ 
88                  $F_{ss} = \langle Act|F|Act \rangle \langle Part|Part \rangle$ 
89             endif; enddo
90
91         if [Hole .and. (M-1).ge.N]
92             do M1N: LM1N
93                  $F_{ss} = \langle Hole|F|Hole \rangle \langle Act|Act \rangle$ 
94                  $(ss|ij) = \langle Hole|V|Hole \rangle \langle Act|Act \rangle$ 
95                  $(si|sj) = \langle Hole|V|Hole \rangle \langle Act|Act \rangle$ 
96             enddo; endif
97         if [Part .and. N.ge.2]
98             do M1N2: LM1N2
99                  $(ss|ab) = \langle Part|V|Part \rangle \langle Act|Act \rangle$ 
100                 $(sa|sb) = \langle Part|V|Part \rangle \langle Act|Act \rangle$ 
101                 $F_{ss} = \langle Part|F|Part \rangle \langle Act|Act \rangle$ 
102            enddo; endif
103
104     endif ! [i1.EQ.i2]
105
106     do i3 = 1,i1

```

```

107 do i4 = 1,i4max
108   define: iXssss ,  $I_{ijkl}$ 
109   FormXij: Lb , SgnB ; LHb , SgnHb ; LPb , SgnPb
110
111 do La
112   do Lb
113     ( $ss|ss$ ) = < Act|V|Act > < Act|V|Act >
114   enddo
115   if [Hole .and. MNHb .gt .0]
116     do LHb
117       ( $ss|ss$ ) = < Hole|V|Hole > < Act|V|Act >
118     enddo; endif
119   if [Part .and. MNPb .gt .0]
120     do LPb
121       ( $ss|ss$ ) = < Part|V|Part > < Act|V|Act >
122     enddo; endif; enddo
123
124 if [ $I_{ijk}$  .NE. 1 , 7 .and. Hole .or. Part ]
125   do Lb
126     if [Hole .and. MNHa .gt .0]
127       do LHa
128         ( $ss|ss$ ) = < Hole|V|Hole > < Act|V|Act >
129       enddo; endif
130     if [Part .and. MNPa .gt .0]
131       do LPa
132         ( $ss|ss$ ) = < Part|V|Part > < Act|V|Act >
133       enddo; endif; enddo; endif
134
135 if [ $I_{ijkl} = 1, 3, 5$ ]
136   deallocate LB, LHb, LPb
137   cycle i4
138
139 elseif [ $I_{ijkl} = 2, 7 .and. M .ge. 2$ ]
140   if [N .ge. 2]
141     do LM2N2
142       ( $ss|ss$ ) = < Act|V|Act > < Act|Act >
143       if [Hole]
144         ( $ss|ss$ ) = < Act|V|Act > < Hole|Hole >
145       endif
146       if [Part]
147         ( $ss|ss$ ) = < Act|V|Act > < Part|Part >
148       endif; enddo; endif
149   if [Hole .and. (M-2) .ge. (N-1)]
150     do LM2N1
151       ( $ss|ss$ ) = < Hole|V|Hole > < Act|Act >
152     enddo; endif
153   if [Part .and. N .ge. 3]
154     do LM2N3
155       ( $ss|ss$ ) = < Part|V|Part > < Act|Act >
156     enddo; endif
157
158 elseif [ $I_{ijkl} = 4, 6, 8, 9, 10 .and. M .ge. 3$ ]
159   if [N .ge. 2 .and. (M-3) .ge. (N-2)]
160     do LM3N2
161       ( $ss|ss$ ) = < Act|V|Act > < Act|Act >

```

```

162         if [Hole]
163             (ss|ss) = < Act|V|Act > < Hole|Hole >
164         endif
165         if [Part]
166             (ss|ss) = < Act|V|Act > < Part|Part >
167         endif; enddo; endif
168     if [Hole .and. (M-3).ge.(N-1)]
169         do LM3N1
170             (ss|ss) = < Hole|V|Hole > < Act|Act >
171         enddo; endif
172     if [Part .and. N.ge.3]
173         do LM3N3
174             (ss|ss) = < Part|V|Part > < Act|Act >
175         enddo; endif
176
177 elseif [Iijkl = 11 .and. M.ge.4]
178     if [N.ge.2 .and. (M-4).ge.(N-2)]
179         do LM4N2
180             (ss|ss) = < Act|V|Act > < Act|Act > (2 times)
181             if [Hole]
182                 (ss|ss) = < Act|V|Act > < Hole|Hole > (2 times)
183             endif
184             if [Part]
185                 (ss|ss) = < Act|V|Act > < Part|Part > (2 times)
186             endif; enddo; endif
187
188         if [Hole .and. (M-4).ge.(N-1)]
189             do LM4N1
190                 (ss|ss) = < Hole|V|Hole > < Act|Act > (2 times)
191             enddo
192
193             if [Part .and. N.ge.3 .and. (M-4).ge.(N-3)]
194                 do LM4N3
195                     (ss|ss) = < Part|V|Part > < Act|Act > (2 times)
196                 enddo; endif
197
198             deallocate Lb, LHb, LPb
199         enddo ! i4, i3
200
201     if [Hole .or. Part]
202         do i3 = 1,M
203             if [N.lt.1] cycle i3
204
205             do La(MNa)
206                 if [Hole .and. (M-1).ge.N]
207                     do LM1N
208                         (ss|si) = < Act|V|Hole > < Act|V|Act >
209                         if [Iijkl.ge.9]
210                             (ss|si) = < Act|V|Hole > < Act|V|Act >
211                         endif; enddo; endif
212                     if [Part]
213                         do LM1N1
214                             (ss|sa) = < Act|V|Part > < Act|V|Act >
215                             if [Iijkl.ge.9]
216                                 (ss|sa) = < Act|V|Part > < Act|V|Act >

```

```

217     endif; enddo; endif; enddo
218
219     NMin = 1; IF [.NOT.Hole] NMin = 2
220     if [N.lt.NMin .or. Iijkl = 5]
221         cycle i3
222
223     elseif [Iijkl = 6,9,10]
224         if [Hole .and. (M-2).ge.(N-1)]
225             do LM2N1
226                 (ss|si) = < Act|V|Hole > < Act|V|Act >
227             enddo; endif
228         if [Part .and. N.ge.2]
229             do LM2N2
230                 (ss|sa) = < Act|V|Part > < Act|V|Act >
231             enddo; endif
232
233     elseif [Iijkl = 11 .and. M.ge.3]
234         if [Hole .and. (M-3).ge.(N-1)]
235             do LM3N1
236                 (ss|si) = < Act|V|Hole > < Act|V|Act > (2 times)
237             enddo; endif
238         if [Part .and. N.ge.2 .and. (M-3).ge.(N-2)]
239             do LM3N2
240                 (ss|sa) = < Act|V|Part > < Act|V|Act > (2 times)
241             enddo; endif
242
243     enddo ! i3, i2, i1
244
245     if [Part]
246         Fab = < Part|F|Part > < Act|Act >
247     endif
248     if [Hole]
249         Fij = < Hole|F|Hole > < Act|Act >
250     endif
251
252     write response vector (jR) into disk
253
254 end Roots

```