A New Fortran Program for CFP's
of a Identical Fermion System

$S\omega 9443$

Jia-jun Wang

Qi-zhi Han

Yu-xin Liu

P.O.Box 2735, Beijing 100080, The People's Republic of China

Telefax : (86)-1-2562587   Telephone : 2568348

Telex : 22040 BAOAS CN   Cable : 6158

A New Fortran Program for CFP's of a Identical Fermion System

Jia-jun Wang

Department of Physics, Peking University, Beijing 100871, China

Institute of Theoretical Physics, Academia Sinica, Beijing 100080, China

Qi-zhi Han

Department of Physics, Peking University, Beijing 100871, China

Yu-xin Liu

CCAST(World Lab), P. O. Box 8730, Beijing 100080, China

Department of Physics, Peking University, Beijing 100871, China

Institute of Theoretical Physics, Academia Sinica, Beijing 100080, China

Abstract

A new Fortran program called CFPSIF is introduced. Using the new recurrent formula obtained by us the program CFPSIF calculates the fractional parentage coefficients (CFPs) of identical fermion system with well-defined seniority. The recurrent process is controlled by the multiplicity of the irreducible representations(IRREPs) of group $O(3)$ in the reduction $SP(N) \supset O(3)$. It is shown that the program CFPSIF is more efficient than the program GENESIS, especially for the system with a relatively large number of fermions.

PROGRAM SUMMARY

Nature of physical problem

The program calculates all the coefficients of fractional parentage (CFPs)[1] of identical fermion system, by using the recurrent relations [2] with well-defined seniority. It is of fundamental importance in constructing a many-particle wave function with well-defined permutational symmetry and total angular momentum CFPs method is one of the most efficient method for construction of such wave functions. It is an iterative method, namely, the n-particle wave function is constructed from (n-1) particle wave functions by coupling one more particle. The reduced matrix elements of any physical tensor operator can be calculated easily with the wave functions expressed in terms of the CFPs. Therefore the calculation of CFPs plays an important role in nuclear physics, atomic and molecular physics.

Method of solution

The program CFPSIF uses the approach developed in Refs.[2] to get all the CFP's of an identical fermion system. In the approach, the CFP is factorized as a product of the isoscalar factor(ISF) of the reduction $U(N) \supset SP(N)$ and that of $SP(N) \supset O(3)$. The ISF of the reduction $U(N) \supset SP(N)$ has been given analytically. The ISF of the reduction $SP(N) \supset O(3)$ is evaluated by a recurrent relation. The recurrent relation is presented with well-defined seniority, and the recurrent process is controlled by the multiplicity of an irreducible representation (IRREP) of O(3) in an IRREP of group SP(N). It provides an efficient algorithm for computation and is numerically stable for relatively large system.

Restriction on the complexity of the problem

The program can evaluate the CFPs of a system including identical fermions, each with single angular momentum j. At present, with the dimension control parameters MNU=6, NFS=40, NFJ=20, the program can handle the system with $j \leq 11/2$. If we take the parameters MNU=3, NFS=300, NFJ=40, the program can handle the system with $j \leq 15/2$. After the parameters MNU, NFS, NFJ et al. are enlarged, the system can be enlarged.

Typical running time

This depends strongly on the fermion number and the single angular momentum . For example, it takes about 67 minutes on VAX 8550 to get all the CFPs of a fermion system with $j = 15/2$, but it takes only about 3 seconds for the fermion system with $j = 9/2$.

References

[1] R. F. Bacher and S. Gousmit, *Phys. Rev.* 46 (1934) 948;

G. Racah, *Phys. Rev.* 63 (1943) 367;

A. R. Edmonds and B. H. Flowers, *Proc. R. Soc. London* A 214 (1952) 515;

P. J. Redmond, *Proc. R. Soc. London* A 222 (1954) 84;

A. DeShalit and I, Talmi, *Nuclear Shell Theory* (Academic, New York, 1963).

[2] Hong-zhou Sun, Qi-zhi Han, Mei Zhang and Gui-lu Long, *Commun. Theor. Phys.* 11 (1989) 441.

[3] Jia-jun Wang, Hong-zhou Sun. *High Energy Phys. Nucl.Phys,(in Chinese)* 14 (1990) 842.

LONG WRITE–UP

1. Introduction

It is well known that the fractional parentage coefficients (CFPs)[1-2] method is one of the most efficient method for the construction of shell model wave functions. It is an iterative method, namely, the n-particle wave function is constructed from $(n - 1)$-particle wave functions by coupling one more particle. An over-complete nonorthogonal set of n-particle wave function can be formed. Then one has to find out the linear independent and orthonormalized wave function by using an appropriate orthogonalization method. Since then many works have been done[3-6] to improve the evaluation procedure, and some computer codes, such as JJCFP[7], GENESIS[8,9], CFPSIB[10], etc have also been set up to evaluate the CFPs. However, all these codes are not efficient enough for the system including a large number of identical particles each with higher single j. There are two reasons for the limitation. The first is that the recurrent formulae of CFPs were not represented with explicit seniority. The ones with definite seniority should then be calculated by diagonalizing the quadratic Casimir operators of group $U(N)$ and group $SP(N)$ (for fermions) or $O(N)$ (for bosons) simultaneously, repeating calculations increase rapidly with the increasing of the particle number n and the single angular momentum j. The second is that the rounding errors of numerical calculations in the procedure of choosing the linear independent CFPs from the over-complete set are unavoided. On other hand, In order to investigate the physical mechanism of high spin states and superdeforemation in nucleus and to examine the chaotic behaviour in many fermion system, it is imperative to develop a computer code for evaluating CFPs of the system including a large number of fermions, each with higher angular momentum(eg. $j = 13/2$, $15/2$, $17/2$, $\cdots\cdots$).

Recently, Hong-Zhou Sun and the collaborators have developed a new recurrent relations for determining the CFPs of fermion system with explicit seniority[6]. In the new approach, the wave functions are classified according to the group chain $U(N) \supset SP(N) \supset O(3)$ ($N = 2j + 1$), and the CFPs are factorized into the product of the isoscalar factors (ISFs) of the group chains $U(N) \supset SP(N)$ and $SP(N) \supset O(3)$. The ISFs of $U(N) \supset SP(N)$ have been obtained analytically. The ISFs of $SP(N) \supset O(3)$ can be calculated by the recurrent formula with explicit seniority[6]. It has been shown that this approach is quite efficient for calculating the CFPs of the system including a relatively large number of fermions, each with higher j, and the method is numerically stable for large system. In the framework of this new approach, we set up a FORTRAN program CFPSIF. With this code, the CFPs of a system including a large number of fermions, each with a physically meaningful angular momentum j can be determined efficiently.

In this paper, the program is introduced in details. In section 2, we give briefly the mathematical formulae of the recurrent relations and the method of calculation. In section 3, the program structure is described. Finally, we show the results of test runs and give a brief discussion on the program.

2. Mathematical Formulae and Calculation Method

In the program CFPSIF, the new recurrent relations of CFPs[6] and the formulae of multiplicities(MULs)[6,11] are used. The CFPs are factorized as

$$
\langle n\ \nu\ \alpha\ J \{ | n - 1\ \nu_1\ \alpha_1\ J_1\ j\ J \rangle
$$
$$
= \langle n\ \nu\ \alpha\ J\ \|a_j^\dagger\| n - 1\ \nu_1\ \alpha_1\ J_1 \rangle / \sqrt{n}
$$
$$
= \left\langle \begin{matrix} [1] & [1^{n-1}] \\ (1) & (1^{\nu_1}) \end{matrix} \middle| \begin{matrix} [1^n] \\ (1^\nu) \end{matrix} \right\rangle \left\langle \begin{matrix} (1) & (1^{\nu_1}) \\ j & \alpha_1\ J_1 \end{matrix} \middle| \begin{matrix} (1^\nu) \\ \alpha\ J \end{matrix} \right\rangle , \tag{1}
$$

in which $|n\ \nu\ \alpha\ J\rangle$ is the state with fermion number n, seniority ν, total angular momentum J and additional quantum number α. $\left\langle \begin{matrix} [1] & [1^{n-1}] \\ (1) & (1^{\nu_1}) \end{matrix} \middle| \begin{matrix} [1^n] \\ (1^\nu) \end{matrix} \right\rangle$ and $\left\langle \begin{matrix} (1) & (1^{\nu_1}) \\ j & \alpha_1\ J_1 \end{matrix} \middle| \begin{matrix} (1^\nu) \\ \alpha\ J \end{matrix} \right\rangle$ are the ISFs of group reduction $U(N) \supset SP(N)$ and the ISFs of $SP(N) \supset O(3)$ respectively, in which $\nu_1 = \nu - 1$ or $\nu_1 = \nu + 1$. The ISFs of

$U(N) \supset SP(N)$ in the two cases have analytically been given as[6]

$$\left\langle \begin{array}{cc} [1] & [1^{n-1}] \\ (1) & (1^{\nu-1}) \end{array} \middle| \begin{array}{c} [1^n] \\ (1^\nu) \end{array} \right\rangle = \sqrt{\frac{\nu(\nu - n - \nu + 2)}{n(N - 2\nu + 2)}} \qquad (2)$$

$$\left\langle \begin{array}{cc} [1] & [1^{n-1}] \\ (1) & (1^{\nu+1}) \end{array} \middle| \begin{array}{c} [1^n] \\ (1^\nu) \end{array} \right\rangle = -\sqrt{\frac{(\nu - 1)(N - \nu + 2)}{n(N - 2\nu + 2)}} . \qquad (3)$$

So that the key point in calculating the CFPs becomes the evaluation of the ISFs $\left\langle \begin{array}{cc} (1) & (1^n) \\ j & \alpha_1 J_1 \end{array} \middle| \begin{array}{c} (1^\nu) \\ \alpha J \end{array} \right\rangle$ of the reduction $SP(N) \supset O(3)$. The ISFs of $SP(N) \supset O(3)$ with $\nu_1 = \nu + 1$ and $\nu_1 = \nu - 1$ hold reciprocal relation

$$\left\langle \begin{array}{cc} (1) & (1^{\nu+1}) \\ j & \alpha_1 J_1 \end{array} \middle| \begin{array}{c} (1^\nu) \\ \alpha J \end{array} \right\rangle = (-)^{J-j-J_1} \sqrt{\frac{(2J_1 + 1)d(N,\nu)}{(2J + 1)d(N,\nu+1)}} \left\langle \begin{array}{cc} (1) & (1^\nu) \\ j & \alpha J \end{array} \middle| \begin{array}{c} (1^{\nu+1}) \\ \alpha_1 J_1 \end{array} \right\rangle , \qquad (4)$$

where

$$d(N, \nu) = \frac{(N + 2 - 2\nu)(N + 1)!}{\nu!(N + 2 - \nu)!} .$$

The ISFs of $SP(N) \supset O(3)$ with $\nu_1 = \nu - 1$ can be obtained with the following recurrent formula[6]

$$\left\langle \begin{array}{cc} (1) & (1^{\nu-1}) \\ j & \alpha_1 J_1 \end{array} \middle| \begin{array}{c} (1^\nu) \\ (\alpha_1' J_1') J \end{array} \right\rangle = \frac{P(\alpha_1' J_1' \alpha_1 J_1 J)}{\sqrt{\nu \, P(\alpha_1' J_1' \alpha_1' J_1' J)}} , \qquad (5)$$

where $P(\alpha_1' J_1' \alpha_1 J_1 J)$ is given as

$$P(\alpha_1' J_1' \alpha_1 J_1 J) = \delta_{\alpha_1' \, \alpha_1} \, \delta_{J_1' \, J_1}$$
$$+ (-)^{J+J_1'} (\nu-1)\sqrt{(2J_1' + 1)(2J_1 + 1)} \sum_{\alpha_2, J_2} \left[\left\{ \begin{array}{ccc} j & J_2 & J_1' \\ j & J & J_1 \end{array} \right\} + \frac{2\delta_{J_2 \, J}}{(2J+1)} \frac{(-1)^\nu}{N+4-2\nu} \right]$$
$$\left\langle \begin{array}{cc} (1) & (1^{\nu-2}) \\ j & \alpha_2 J_2 \end{array} \middle| \begin{array}{c} (1^{\nu-1}) \\ \alpha_1 J_1 \end{array} \right\rangle \left\langle \begin{array}{cc} (1) & (1^{\nu-2}) \\ j & \alpha_2 J_2 \end{array} \middle| \begin{array}{c} (1^{\nu-1}) \\ \alpha_1' J_1' \end{array} \right\rangle . \qquad (6)$$

From eqs (4-6) we know that all the ISFs of $SP(N) \supset O(3)$ with $\nu \leq n$ can be obtained with initial value $\left\langle \begin{array}{cc} (1) & (1) \\ j & J_1 \end{array} \middle| \begin{array}{c} (1^2) \\ J \end{array} \right\rangle = 1 \; (J = 0, 2, 4, \cdots, 2j - 1)$. Of course

the states $|\nu \; \nu \; (\alpha_1' \; J_1') \; J \; M\rangle$ constructed in this way are over-complete. We then have to find out the linear independent and orthonormal states $|\nu \; \nu \; \alpha \; J \; M\rangle$ and the corresponding ISFs $\left\langle \begin{array}{cc} (1) & (1^{\nu-1}) \\ j & \alpha_1 J_1 \end{array} \middle| \begin{array}{c} (1^\nu) \\ \alpha J \end{array} \right\rangle$ by taking the advantage of the Schmit orthogonalization method.

In order to make the orthogonal process more efficient, we use the MULs of states with the same ν and J to control the calculation. The MULs of the total angular momentum $J = J_{max} - \xi$ of single-j fermion system is[11]

$$\gamma(\nu \, J) = \beta(\nu \, J) - \beta(\nu - 2 \, J), \qquad (7)$$

in which

$$\beta(\nu \, J) = \sum_{\alpha=1}^{\nu} P_{\alpha \, m}(\xi) - \sum_{\alpha=1}^{\nu} P_{\alpha \, m}(\xi - 1), \qquad (8)$$

where $P_{\alpha \, m}(\xi)$ is the α-row partition of an integer $\xi \geq 0$, it is equal to the number of states with quantum number

$$m = 2j + 1 - \nu \geq 0$$

and

$$J_{max} = \frac{1}{2}\nu(2j + 1 - \nu).$$

The $P_{\alpha \, m}(\xi)$ can be calculated by a recurrent formula[11]

$$P_{\alpha \, m}(\xi) = \sum_{i=1}^{\alpha-1} \sum_{j=i}^{F(\xi)} P_{i \, m-j}(\xi - \alpha \, j) \qquad (9)$$

$$F(\xi) = \begin{cases} [\xi/\alpha], & \xi \geq 2\alpha, \\ 1, & \xi < 2\alpha \end{cases},$$

where $[\xi]$ indicates the integer part of number ξ (e.g., $[3/2] = 1$). $P_{1 \, m}(\xi)$ and $P_{2 \, m}(\xi)$ are initial values of $P_{\alpha \, m}(\xi)$[11].

3. Program Structure

CFPSIF is a program written in FORTRAN by exploiting the new recurrent formula[6] to get all the CFPs of a system with n identical fermions, each with angular momentum j. It consists of the main programs CFPSIF, subroutines MULT, INMULT, FISF, GEN, INISF, CFP, QUIT and some functions. The relation between

the notations in the program and those in the formulae in section 2 are shown in table 1.

Table 1 Relation between notations in the text and CFPSIF

In text		In CFPSIF
$\left\langle \begin{array}{cc} (1) & (1^n) \\ j & \alpha_1 J_1 \end{array} \right. \left| \begin{array}{c} (1^\nu) \\ (\alpha_1' J_1') J \end{array} \right\rangle$		PGISF(NU,K0,KP,JD)
$P(\alpha_1' J_1' \alpha_1 J_1 J)$		P(NU,K0,KP,JD)
$\gamma(\nu\, J)$		MULT
$\beta(\nu\, J)$		BETA(NV,J,SIJ)
$P_{\alpha\, m}(\xi)$		PF(K,M,X)

where SIJ is the double value of the single angular momentum j. NU and NV are the seniority ν. JD and J are the angular momentum of daughter states and K0, KP are the sequence number of parent states (with seniority $\nu - 1$). In function PF(K,M,X), K refers to α, M reads m and X represents ξ.

3.1 Main Program CFPSIF

After some dimension control parameters are given at the beginning of the program, the single angular momentum j and the particle number n are input. Then, the program call the subroutine MULT and FISF to calculate all the MULs of the states with explicit seniority and the ISFs $\left\langle \begin{array}{cc} (1) & (\nu-1) \\ l & \alpha_1 J_1 \end{array} \right. \left| \begin{array}{c} (\nu) \\ \alpha\, J \end{array} \right\rangle$ with $\nu = n, n-1, n-2, \cdots, 2, 1$. There are some data files to store the calculated results and several sentences controlling repetitious calculations are contained in every subroutine. At last the main program call subroutine CFP to get all the reduced matrix elements (RMEs) and the CFPs with explicit seniority for n fermions in j-j coupling scheme.

3.2 Subroutine MULT

Subroutine MULT employs eq. (7) to get all the MULs of the states for a fermion system, each with angular momentum j and maximal seniority ν_{max}. The states can be the ones with seniority $\nu = \nu_{max}, \nu_{max} - 1, \nu_{max} - 2, \cdots, 2, 1, 0$ and angular momentum

$J = J_{max} - \xi$ $(0 \le \xi \le J_{max})$, where $J_{max} = \frac{1}{2}\nu(2j + 1 - \nu)$. The subroutine MULT comprises functions BETA(NV,J,SIJ),PEFE(N,X,SIJ) and PF(K,M,X). Function BETA(NV, J, SIL) performs the calculation expressed in eq. (8). Function PEFE(NV, X, SIJ) give the value of $\sum_{k=1}^{N} P_{k\,m}(X)$, where the $P_{k\,m}(X)$ is given by another function PF(K, M, X). In order that the data can be transferred easily, the calculating results are written in the common blocks MUL1 and MUL2. In these common blocks, the parameters MNU, NFS, NFJ, etc are the control parameters which determine the capability of the code. Their values are given at the beginning of the main program. In the common block MUL1, the array INUMU(0:MNU) is the number of the states with different total angular momentum but the same seniority ν. The total angular momentum J of the states with seniority $\nu \le$ MNU and their MULs are stored in the array JMU(0:MNU,NFJ) and ALMU(0:MNU,NFJ) respectively, NFJ is the sequence number of total angular momentum J. In the common block MUL2, JS(0:MNU,NFS) is the same as JMU, but the NFS is the sequence number of states which includes MULs. ALS(0:MNU,NFS) is the additional quantum number α $(0 \le \alpha \le$ ALMU). INUS(0:MNU) is the total number of states with seniority ν and any possible angular momentum JMU. IDREC(0:MNU) is the record index of the direct data file ISFF2 to indicate the position of the ISFs . All the calculated results are stored in the data file MULTF, so that a lot of repeated evaluations are avoided.

3.3 Subroutine FISF

The subroutine FISF gives ISFs of the group reduction $SP(N) \supset O(3)$. The calculation is achieved with the DO loop for variable NU from 3 to MAXNU. The FISF call subroutine INISF(NUP) to give the initial values of ISFs $\left\langle \begin{array}{cc} (1) & (1^{\nu-2}) \\ j & \alpha_2 J_2 \end{array} \right. \left| \begin{array}{c} (1^{\nu-1}) \\ \alpha_1 J_1 \end{array} \right\rangle$ with all the possible $\alpha_2, J_2, \alpha_1, J_1$. The argument NUP is the seniority $\nu - 1$. FISF call subroutine GEN(NU,KDJ) to calculate the orthonormalized ISFs. The arguments NU,KDJ correspond to seniority ν and the sequence number of total angular momentum. The total angular momentum J and their MULs can be found out from the common block MUL1. In the subroutine GEN(NU,KDJ), the additional quantum number α $(0 \le \alpha \le$

ALMU) are denoted by A, which is a cyclic variable. The nonorthogonal ISFs are calculated by the function PGISF(NU,K0,KP,JD) and are stored in the array XX(KP,A) at first. The function PGISF employs eq (4) and comprises the function P(NU,K0,KP,JD) which employs eq (5). The orthogonalized ISFs are then given with the Schmidt orthogonalizing method and stored in array OISF(KP,A). In order to make the orthogonal process more efficient,we use the MULs to control it and we set up an accuracy argument SUACC to select appropriate K0 for giving the orthonormalized ISFs. Meanwhile the orthonormality of the calculated results are examined at every step of the evaluation. If OISF(KP,A) are nonorthogonal or nonnormal, the GEN call the subroutine QUIT to stop the calculation and give the information "error in orthogonality" or "error in normalizing". The Subroutine QUIT terminates the program as any kind of running error takes place and gives the error information, such as, error in orthogonalizing, error in normalizing, IDR is inconsistent, and so on. Finaly the ISFs are stored in a data file ISFF2. It is a direct data file.

3.4 Subroutine CFP

Subroutine CFP gives the CFPs $\langle n\ \nu\ \alpha\ J\ \| n-1\ \nu_1\ \alpha_1\ J_1\ j\ J\rangle$ and the RMEs $\langle n\ \nu\ \alpha\ J\ \|a^1\|\ n-1\ \nu_1\ \alpha_1\ J_1\ j\ J\rangle$ by carrying out the calculation shown in eq. (1). The ISFs of $SP(N) \supset O(3)$ $\left\langle \begin{matrix} (1) & (1^{\nu-1}) \\ j & \alpha_1\ J_1 \end{matrix} \middle| \begin{matrix} (1^\nu) \\ \alpha\ J \end{matrix} \right\rangle$ are read from data file ISFF2. The ISFs of $U(N) \supset SP(N)$ are given analytically by eq. (2),(3). Whether the calculated results are normalized are also checked. If there are error in normalizing, the subroutine CFP call QUIT to give the information too. The calculated results are written in data files CFPOUT and RMEOUT respectively.

4. Results and Discussion

As mentioned above the input card and the operation of this program are quite simple. After the command RUN CFPSIF, one can input the double value of the single angular momentum $2j$ and the maximal seniority number ν of the fermions along the line suggested by the test run information about the input data, For instance, 15, 3. Then,

several other items, such as the range of the fermion number and the information about whether the MUL data and the ISF data have existed or not should also be input (see the first part of the TEST RUN OUTPUT). Thereafter, all the RMEs and CFPs with all the possible fermion numbers can be obtained. The part 2 and part 3 of the TEST RUN OUTPUT are the examples of the calculated RMEs and CFPs. These two cards show that the output data of this program are physically meaningful and explicit, which can be used directly for performing other calculations.

To show the capability of the code, we run the codes CFPSIF and GENESIS on the computer VAX-8550 to calculate all the CFPs of several identical fermion system. It takes about 67 minutes to accomplish the calculation for the system with $j = 15/2$ and $n = 8, 7, \cdots, 3$ with the code CFPSIF. The comparison between the running times for accomplishing the calculations for the other system with CFPSIF and that with GENESIS is shown in table 2.

Table 2 the running time on VAX-8550 for several fermion system

j name of code	9/2	11/2	13/2
CFPSIF	3 sec.	13 sec.	3 min.
GENESIS	6 min.	33 min.	37 min.*

* It is terminated when the CFPs with $n = 5$, $\nu = 5$, $J = 45/2$ are being evaluated.

The table indicates that, on one hand, the calculation speed of the code CFPSIF depends strongly on the number of fermions and the single angular momentum of each fermion. On the other hand, the code CFPSIF is much more efficient than the code GENESIS. The reason for this is that, in the new approach, the recurrent formulae are given with well-defined seniority ν and the multiplicity of a L in an IRREP of group $SP(N)$ are used to control the orthogonal process. Whereas GENESIS diagonalizes the Casimir operators of $SP(N)$ to obtain definite ν. A lot of repeatedly computations are then avoided in this new code. Because the dimension of $< 1^\nu >$ is much smaller than

that of $[1^n]$, this new formalism saves computer space distinctly. Moreover. this new algorithm can decrease rounding errors efficiently. These merits of the new calculating method[6] make the code CFPSIF very efficient. Therefore, this code can be used to treat relatively large system of fermions, each with a meaningful angular momentum j. At present, with the parameters given in the program, the code can be used for the system with $j = 15/2$ and seniority $\nu = 8$, The capability can be strengthened by enlarging the parameters MNU, NFJ, NFS, MAL and MM, if it is needed. The values MNU, NFJ, MAL and MM can be given as what is required. The economic value of NFS can be determined by running the subroutine MULT separately with a quite large trying NFS.

References

[1] R.F. Bacher and S. Goudsmit, *Phys. Rev.* 46 (1934) 948 .

[2] G. Racah, *Phys. Rev.* 63 (1943) 367.

[3] B. F. Bayman and A. Lande, *Nucl. Phys.* 77 (1966) 1.

[4] A. DeShalit and I, Talmi, *Nuclear Shell Theory* (Academic, New York, 1963).

[5] A. Novoselsky, J. Katriel and R. Gilmore, *J. Math. Phys.* 29 (1988) 1368.
   A. Novoselsky, M. Vallieres and R. Gilmore, *Phys. Rev.* C 38 (1988), 1440.

[6] H. Z. Sun, Q.Z. Han, M. Zhang and G.L. Long, *Commun. Theor. Phys.* 11 (1989) 441.

[7] L. B. Hubbard, *Comput. Phys. Commun.* 1 (1970) 225.

[8] D. Zwart, *Compt. Phys. Commun.* 38 (1985) 365.

[9] X. Ji and M. Vallieres, *Phys. Rev.* C35 (1987) 1583.

[10] Yu-xin Liu, Hong-zhou Sun and En-guang Zhao, *Compt. Phys. Commun.* 70 (1992) 154.

[11] J.J.Wang and H.Z.Sun. *High Energy Phys. Nucl.Phys, (in Chinese)* 14, (1990) 842.

TEST RUN OUTPUT

Part 1. Examples of the test run output about the input data and running process

```
$RUN CFPSIF
Please input the double value of the single particle angular momentum SIJ
and maximal seniority MAXNU which will be calculated at this time
15,8
Please input minimal number of particles NMIN and maximal number
of particles NMAX
3,8
Is there date file MULTF ? (T or F)
F
Begin to calculate the multiplicity
begin to write the date to file MULTF
The end of mult
The subroutine MULT has been called
Is there date file ISFF2 ? (T or F)
F
end of program
$
   LYX          job terminated at  8-JUN-1994 23:27:21.41

  Accounting information:
  Buffered I/O count:      338    Peak working set size:  1907
  Direct I/O count:       1850    Peak page file size:    4339
  Page faults:            7941    Mounted volumes:           0
  Charged CPU time:    0 01:00:00.35  Elapsed time:   0 06:07:37.00

$RUN CFPSIF
Please input the double value of single particle angular momentum SIJ
and maximal seniority MAXNU which will be calculated at this time
15,8
Please input minimal number of particles NMIN and
maximal number of particles NMAX
6,6
Is there date file MULTF ? (T or F)
t
SIJ=15  maximum number of J= 31
maximum number of state = 278  maximum multiplicity =17
The multiplicities are read
Is there date file ISFF2 ? (T or F)
t
The ISFF2 has already existed
The file ISFF2 is opened
end of program
$
   LYX          job terminated at  8-JUL-1994 11:28:42.77

  Accounting information:
  Buffered I/O count:       511    Peak working set size:  1221
  Direct I/O count:        1733    Peak page file size:    1559
  Page faults:             5403    Mounted volumes:           0
  Charged CPU time:    0 00:05:13.30  Elapsed time:   0 00:04:40.24
```

15

Part 2. Parts of the test run output about the calculated reduced matrix elements

n	nu	alpha	2J	||a+||	nu-1	alpha'	2J'	2j	
< 6	6	1	0	||a+||	5	1	15	15>=	0.0000000000000122
....									
< 6	6	1	2	||a+||	5	1	17	15>=	-0.0906484648585320
< 6	6	1	2	||a+||	5	2	17	15>=	-0.6166009306907654
....									
< 6	6	1	2	||a+||	5	9	17	15>=	-0.7357733249664307
< 6	6	1	2	||a+||	5	10	17	15>=	0.0180810783058405
....									
< 6	6	5	24	||a+||	5	2	31	15>=	0.0664481148123741
< 6	6	5	24	||a+||	5	3	31	15>=	0.0728428736329079
....									
< 6	6	1	40	||a+||	5	1	51	15>=	0.3180595040321350
< 6	6	1	40	||a+||	5	1	55	15>=	-0.7084957957267761
....									
< 6	6	2	40	||a+||	5	8	27	15>=	-0.067013848476791
< 6	6	2	40	||a+||	5	9	27	15>=	0.0063921869732440
....									
< 6	6	3	40	||a+||	5	1	37	15>=	0.2132312059402466
< 6	6	3	40	||a+||	5	2	37	15>=	-0.1698909401893616
....									
< 6	6	4	40	||a+||	5	2	27	15>=	-0.0289981719106436
< 6	6	4	40	||a+||	5	3	27	15>=	0.0407621413469315
....									
< 6	6	5	40	||a+||	5	1	33	15>=	0.4567079842090607
< 6	6	5	40	||a+||	5	2	33	15>=	0.1599017530679703
....									
< 6	6	6	40	||a+||	5	5	31	15>=	-0.2280130982398987
< 6	6	6	40	||a+||	5	6	31	15>=	-0.0624762214720249
....									
< 6	6	7	40	||a+||	5	7	31	15>=	0.4075197279453278
< 6	6	7	40	||a+||	5	8	31	15>=	-0.2483996599912643
....									
< 6	4	1	10	||a+||	3	2	17	15>=	0.4362092614173889
< 6	4	1	10	||a+||	3	1	19	15>=	0.1099684834480286
....									
< 6	4	1	44	||a+||	3	1	29	15>=	0.6744173169136047
....									
< 6	4	1	48	||a+||	3	1	35	15>=	-0.8495907783508301
< 6	4	1	48	||a+||	3	1	39	15>=	1.3908867835998535
....									
< 6	4	1	20	||a+||	3	2	17	15>=	-0.1195470914244652
< 6	4	2	20	||a+||	3	2	17	15>=	-0.0228573381900787
< 6	4	3	20	||a+||	3	2	17	15>=	0.178751319468353
< 6	4	4	20	||a+||	3	2	17	15>=	0.313544955730438
< 6	4	5	20	||a+||	3	2	17	15>=	-0.0782443583011627
< 6	4	6	20	||a+||	3	2	17	15>=	0.0253780130296948

Part 3. Parts of the test run output about the calculated coefficient of fractional

parentage

```
N= 6   NU= 6   NU-1= 5
    n  nu  alpha  2J     nu-1 alpha'2J'  2j
<   6   6    1     2 !}   5    1      13  15>=   0.0432644747197628
. . . . . . . . . . . . . . . . . . . . . . . . . . . .
<   6   6    2     2 !}   5   10      17  15>=   0.2110424339771271
. . . . . . . . . . . . . . . . . . . . . . . . . . . .
<   6   6    1     4 !}   5   11      19  15>=  -0.0031551020219922
. . . . . . . . . . . . . . . . . . . . . . . . . . .
<   6   6    4     4 !}   5   11      19  15>=   0.0322970673441887
. . . . . . . . . . . . . . . . . . . . . . . . .
<   6   6    7     4 !}   5    9      19  15>=   0.3094967305660248
<   6   6    7     4 !}   5   10      19  15>=  -0.0170736480504274
<   6   6    7     4 !}   5   11      19  15>=   0.0866909697651863
. . . . . . . . . .
<   6   6    1    40!}    5    4      25  15>=   0.0190073978155851
<   6   6    1    40!}    5    5      25  15>=  -0.1910710632801056
. . . . . . . . . . . .
<   6   6    2    40!}    5    2      37  15>=   0.0356427989900112
<   6   6    2    40!}    5    3      37  15>=   0.0643436536192894
. . . . . . . . .
<   6   6    3    40!}    5    3      31  15>=   0.4552638828754425
<   6   6    3    40!}    5    4      31  15>=   0.0572922490537167
. . . . . . . . . .
<   6   6    4    40!}    5    2      35  15>=   0.0148607529699802
<   6   6    4    40!}    5    3      35  15>=   0.3135749697685242
. . . . . . . . . .
<   6   6    5    40!}    5    5      33  15>=   0.1150218471884727
<   6   6    5    40!}    5    6      33  15>=   0.0741826519370079
. . . . . . . . . . . .
<   6   6    6    40!}    5    1      51  15>=  -0.0086210519075394
<   6   6    6    40!}    5    1      53  15>=   0.0001996028004214
. . . . . . . . . . . .
<   6   6    7    40!}    5    1      41  15>=  -0.0724440589547157
<   6   6    7    40!}    5    2      41  15>=   0.0256840679794550
. . . . . . . . . . . .
N= 6   NU= 4   NU-1= 3
<   6   4    1    14!}    3    1       1  15>=  -0.0639095455408096
<   6   4    2    14!}    3    1       1  15>=   0.0233118633429209
<   6   4    3    14!}    3    1       1  15>=   0.0937244097391764
. . . . . . . . . . . .
<   6   4    5    24!}    3    1      11  15>=  -0.0001558394445672
<   6   4    1    26!}    3    1      11  15>=  -0.1203252719085525
<   6   4    2    26!}    3    1      11  15>=   0.0063809876184400
                                                      -- --()
      5   4    4    26!}    3    1      11  15>=  -0.0002342367947579

     5   4    1    52!}    3    3      25  15>=   0.0865022210891115
     5   4    2    52!}    3    3      25  13>=   0.0020930053056079
     5   4    3    52!}    3    3      25  15>=   0.0613669427401994
     5   4    4    52!}    3    3      25  15>=  -0.0613641125241131
```