

CONTINUUM WAVEFUNCTION SOLVER FOR GRASP

W.F. PERGER*, Z. HALABUKA**, AND D. TRAUTMANN***

**Electrical Engineering Department and Physics Department, Michigan Technological University, Houghton, MI 49931, USA*

***Physics Department, University of Fribourg, CH-1700 Fribourg, Switzerland*

****Institut für Physik der Universität Basel, CH-4056 Basel, Switzerland*

Received: 19 August 1992; in revised form:

Abstract

Relativistic V^{N-1} continuum orbitals are solved numerically within the framework of the GRASP code [1]. Normalization is accomplished with a relativistic WKB method or curve fitting, depending on whether the core is an ion or is neutral. In the normalization, the phase shift (relative to hydrogen) is also calculated. The spacing of the radial grid at large radial distances is automatically determined to provide at least 10 grid points per half cycle of the wavefunction. Lagrange multipliers are automatically determined.

NEW VERSION SUMMARY

Title of new version: CONTWVG

Catalogue number:

Program obtainable from: CPC Program Library, Queen's University of Belfast, N. Ireland
(see application form in this issue)

Reference to original program: Catalogue number: AANC 0001 CONTWV;

Title: Relativistic Continuum Wavefunction Solver;

Ref. in CPC: CPC **66** (1991) 392-402.

Authors of original program: W.F. Perger and V. Karighattam

Does the new version supersede the original program:

No, the original program, CONTWV, was written for the original program of Grant, et al, CPC **21** (1980) 207; this version, CONTWVG, is written for the GRASP program, CPC **55** (1989) 425-456.

Licensing provisions: none

Computer for which the new version is designed and others on which it has been tested:

Computers: Sun SPARC Station, IBM 4381, IBM RS6000, VAX 850;

Installations: Michigan Technological University; Universite Fribourg

Operating systems or monitors under which the new version has been tested:

Sun OS 4.1.1, 4.1.2; IBM CMS VM/SP HOP 5.0; IBM AIX 3.1.5; VAX/VMS V5.4

Programming language used in the new version: FORTRAN 77

Memory required to execute with typical data: 16 million (8 bit) words

No. of bits in a word: 8

Number of processors: 1

Has the code been vectorised? No

No. of lines in distributed program, including test data, etc.: 4856

CPC Program Library subprograms used: Cat. no.: ABJN

Additional Keywords:

Nature of physical problem

The relativistic Dirac-Fock equations are set up and solved numerically for continuum wavefunctions within the framework of the GRASP program, K.G. Dyall, et al, CPC **55** (1989) 425-456.

Method of solution

Relativistic atomic wavefunctions are calculated using a central differences method with deferred corrections within the configuration interaction framework of the GRASP program. The grid spacing at large radial distances (the linear region of the two-piece grid) is automatically determined to provide at least 10 grid points per half cycle of the wavefunction. Lagrange multipliers are automatically calculated without additional input.

Reason for the new version

This new version is required for obtaining relativistic continuum orbitals with the new version of the bound state program, GRASP, and to add refinements to the original CONTWV program.

Restrictions on the complexity of the problem

Continuum orbitals for atoms ranging from hydrogen to mercury have been calculated, with up to six Lagrange multipliers and energies ranging from zero to 100 atomic units. Cases outside of these limits will likely also succeed. The program automatically sets

the spacing of the radial grid to provide for at least 10 grid points per half cycle (of the oscillations in the wavefunction). Consequently, the rapid oscillation of the wavefunction for very large energies will likely result in an error message which instructs the user to increase the number of grid points (which is set by the additional string variable, NPC, in the pre-processor string file).

Typical running time

The typical running time is 10 seconds to 1 minute on an IBM RS6000, Model 540.

Unusual features of the program

The program will calculate a virtual state V^{N-1} continuum orbital for a given κ quantum number and dump that orbital to a file. The program uses a relativistic WKB approximation for normalization of the continuum orbital in the presence of a non-neutral core and it uses a curve-fitting procedure for normalization in the case of a neutral core. The spacing of the radial grid at large radial distances, where the two-piece grid is linear, is automatically calculated to provide at least 10 grid points per half cycle. The procedure for the automatic calculation of the Lagrange multipliers has been improved (relative to CONTWV) to achieve better orthogonality between the orbitals. The program allows the user to choose whether or not to calculate the phase shift, thus providing for better computational speed if the phase shift is not required.

LONG WRITE-UP

1 Introduction

This adaptation, CONTWVG, calculates relativistic continuum wavefunctions within the general structure of the multiconfigurational Dirac-Fock GRASP (General-purpose Relativistic Atomic Structure Program) program of Dylla, et al[1]. It requires that the user has generated, from a previous run of GRASP, a set of core orbitals which is then used by CONTWVG to calculate a continuum orbital. Functionally, CONTWVG is similar to its predecessor, CONTWV[2] and this article will focus on the differences, which are primarily improvements to CONTWV normalization and orthogonalization subroutines.

1.1 Details of Solution

The method of solution of the radial one-electron wavefunctions for the continuum orbitals in CONTWVG is similar to the one used in CONTWV, the first difference being that the exchange potentials are now calculated by using a cubic splines interpolation of arrays XP and XQ after a call to subroutine XPOT (of the GRASP code), thereby eliminating subroutines XPOTC and YZKC of CONTWV. The choice of grid and the numerical method of integration remain the same, but the strategy of iteration and exit have been modified to improve convergence. The subroutine controlling the convergence process, SOLVC, now begins with a lower accuracy parameter and then tightens it as the Lagrange multipliers are being adjusted (instead of maintaining a constant value of the accuracy parameter throughout the convergence process). This tends to more effectively “guide” the solution to convergence more rapidly, i.e. with fewer overall iterations. Furthermore, this accuracy parameter is now also passed to the outward integration subroutine, OUTC, because then OUTC will return a wavefunction which is stable to roughly the degree that SOLVC requires. This change has also reduced the number of iterations required in SOLVC but, more importantly, it has addressed a few cases where the error in the wavefunction after integration by OUTC was larger than the accuracy required by SOLVC, which led to no convergence.

1.2 Normalization

The method of normalization has been improved upon in CONTWVG. For the case where the core is an ion, i.e. the continuum electron sees a non-zero electric potential at large radial distances from the nucleus, a relativistic WKB method has been used. Ong and Russek[3] describe the extension of the WKB method for normalization of a relativistic continuum wavefunction (ionic core case). The approach used in CONTWVG is basically identical except that no approximation is made, aside from the use of the WKB approach. The large component of the continuum wavefunction is taken, in the WKB approximation, to be:

$$P = \frac{A (W - V + mc^2)^{1/2}}{(\pi\hbar c^2)^{1/2} (d\phi/dr)^{1/2}} \cos\phi \quad (1)$$

where A is the (to be determined) normalization constant, $W = -\epsilon + mc^2$ (ϵ is chosen to be a negative number for a continuum state within the GRASP code), V is the usual electric potential ($V = -(Z - N)e/r$ where Z is the number of protons and N is the number of electrons screening the continuum electron) and the expression for $d\phi/dr$ is

$$-\left(\frac{d\phi}{dr}\right)^2 + \frac{(W - V)^2 - m^2c^4}{\hbar^2c^2} - \frac{\kappa(\kappa + 1)}{r^2} + \left(\frac{d\phi}{dr}\right)^{1/2} \frac{d^2}{dr^2} \left(\frac{d\phi}{dr}\right)^{-1/2} - (W - V + mc^2)^{1/2} \frac{d^2}{dr^2} (W - V + mc^2)^{-1/2} + \frac{\kappa}{r} \frac{dV/dr}{(W - V + mc^2)} = 0. \quad (2)$$

Note that (2) differs from equation (4) of Ong and Russek so that the amplitude of (1) approaches $(\frac{2}{\pi k})^{1/2}$ at large radial distances. Defining U as

$$U = \frac{A (W - V + mc^2)^{1/2}}{(\pi\hbar c^2)^{1/2} (d\phi/dr)^{1/2}} \sin\phi \quad (3)$$

then taking the derivative of (1) with respect to r, U can be shown to be

$$U = \left(\frac{d\phi}{dr}\right)^{-1} \left(-\frac{dP}{dr} - \frac{1}{2} \frac{dV}{dr} (W - V + mc^2)^{-1} P - \frac{1}{2} \frac{d^2\phi/dr^2}{d\phi/dr} P \right). \quad (4)$$

Combining (1) and (4) to solve for A and ϕ yields:

$$A = c \left(\frac{\pi\hbar}{W - V + mc^2} \frac{d\phi}{dr} (P^2 + U^2) \right)_{r_0}^{1/2} \quad (5)$$

$$\phi(r_0) = \tan^{-1}(U/P)_{r_0}, \quad (6)$$

where the radial point of evaluation of (5) and (6), r_0 , must be well beyond the point where the exchange potential is negligibly small (for small atoms, $Z < 20$, $r_0 \simeq 15$ -20 a.u.; for larger atoms, e.g. cesium, r_0 should be 40 a.u. or larger). The expression for $d\phi/dr$ in (2) can easily be solved iteratively by approximating $d\phi/dr$ by the second and third terms on the left-hand side of (2), then using that estimate in the next iteration for the fourth term on the left-hand side of (2). With only two iterations, $d\phi/dr$ is stable to seven significant figures.

To calculate the phase shift of the numerical wavefunction relative to the hydrogenic one (with identical energy), the algorithm just described is applied twice: once to calculate the phase of the numerical wavefunction, P, at r_0 and once to calculate the phase of the corresponding hydrogenic wavefunction, also at r_0 . The difference of these two values then represents the phase shift. The accurate determination of the relativistic continuum wavefunction for hydrogen is accomplished by a call to function HYD and is by itself problematic because it requires the evaluation of the large component [4, 5]:

$$P(r) = \frac{(W/c^2 + 1)^{1/2} (2kr)^\gamma e^{\pi y/2} |\Gamma(\gamma + iy)|}{2(\pi k)^{1/2} \Gamma(2\gamma + 1)} \left[e^{-ikr+iy} (\gamma + iy) {}_1F_1(\gamma + 1 + iy, 2\gamma + 1; 2ikr) + c.c. \right], \quad (7)$$

which requires the accurate evaluation of the confluent hypergeometric function, ${}_1F_1(\alpha, \beta; z)$, over a wide range of z since k (the momentum) and r (the radial distance) both range over several decades. For this purpose, the ${}_1F_1$ evaluator of Nardin, et al, [6] was used (and is included with CONTWVG) because it was found that there are values of z for which the asymptotic expansion for large z [7]

$$\begin{aligned}
{}_1F_1(\alpha, \beta; z) \approx & \frac{\Gamma(\beta)}{\Gamma(\beta - \alpha)} e^{i\pi\alpha} z^{-\alpha} \left\{ \sum_{n=0}^R \frac{(\alpha)_n (\alpha - \beta + 1)_n}{(-z)^n n!} + \mathcal{O}(|z|^{-(R+1)}) \right\} \\
& + \frac{\Gamma(\beta)}{\Gamma(\alpha)} e^z z^{\alpha-\beta} \left\{ \sum_{n=0}^S \frac{(\beta - \alpha)_n (1 - \alpha)_n}{z^n n!} + \mathcal{O}(|z|^{-(S+1)}) \right\} \quad (8)
\end{aligned}$$

was not reliable. However, the function call HYD employed by CONTWVG does make use of this asymptotic expansion when $kr > 70$ because it is much faster than the ${}_1F_1$ evaluator.

In the case of a neutral core, i.e. the continuum electron experiences a net zero electric potential at large radial distances from the nucleus, a two parameter curve fitting approach is used. This method is also described in detail in a previous article[2].

1.3 Determination of Lagrange multipliers

The method for determining the Lagrange multipliers has been improved from the simple bisection routine used in CONTWV. In CONTWVG, the two most recent estimates for a given Lagrange multiplier and the two corresponding orthogonalities are used to generate the next estimate of the Lagrange multiplier, using a linear extrapolation. This approach, while still simple, has proven to be able to achieve better ultimate orthogonalities, in typically fewer iterations.

2 Program Structure

2.1 Subroutine DATAIN

The program described here was intended to take maximum advantage of the features of the GRASP program. While the details of the modifications required for subroutine DATAIN are different for CONTWVG than for CONTWV, from the user's point of view, they operate in the same way. Therefore, the structure of the input for CONTWVG is identical to that of CONTWV, i.e. the reading of an additional line from the input file for the purpose of declaring the user's intent to calculate a continuum orbital. This is accomplished by a line, preceding the "END" line, with a label "CONT" followed by three numbers. The first number specifies the energy of the continuum orbital; the second number indicates the point in the grid where the grid is to switch from exponential to linear (variable MHALF); the third number designates the file unit identifier where the continuum wavefunction is to be written. An additional feature of CONTWVG is that, if MHALF is equal to zero, then MHALF and HCONT, the spacing in the linear region of the grid, will be automatically chosen in the following manner. The value of the maximum radial grid point from the bound

state wavefunctions will be the maximum grid point for the continuum grid. Furthermore, ten grid points per half cycle (of the continuum wavefunction) in the linear grid region will be used. These two constraints are then used to set the values of MHALF and HCONT; the first sets MHALF and the second sets HCONT. If these constraints require more grid points than are dimensioned (with variable NPC in the preprocessor), then CONTWVG aborts with an appropriate error message.

In order to avoid extensive changes to the input structure of the GRASP program, the strategy adopted to designate the angular character of the orbital is to use the FIX option for the core orbitals while the free orbital, specified in the convention of the GRASP program, is taken as the continuum. As a result, the principal quantum number of the free orbital is discarded. This strategy will be made more clear with the explanation of the sample run which follows.

2.2 Subroutine SCF

The subroutine of the GRASP program requiring the most extensive changes was SCF. The nature of those changes is the same as in CONTWV, except for the different method for determining the exchange potential and the different method for automatically determining the Lagrange multipliers.

2.3 Subroutines SOLVC, EXTRAP, OUTC

SOLVC, EXTRAP, OUTC of CONTWVG are identical to the ones employed in CONTWV and perform the numerical integration of the Dirac-Fock differential equation.

2.4 Subroutines NORMC and NORMC2

The subroutine NORMC performs the energy normalization of the continuum orbital for the neutral core case using the two parameter fitting procedure described earlier[2]; NORMC2 determines the continuum normalization in the case of an ionic core, also described earlier[2].

2.5 Subroutines SPHJ and SPHN

These subroutines are called by NORMC to provide spherical Bessel and Neumann functions, respectively.

2.6 Subroutine MAXMIN

This routine identifies and prints the position along the radial grid and the amplitudes of the maxima and minima of the wavefunction (same as in CONTWV).

2.7 Subroutine ORTHOG and Function RINTC.

These subroutines calculate the various orthogonalities between the continuum orbital and the core (same as in CONTWV).

2.8 Subroutine SIMP.

This subroutine is a Simpson's integration package (same as in CONTWV).

2.9 Subroutines SPLINE, SPLIN, SPLINT, SPLIND

These subroutines calculate, for the exponential plus linear radial grid, cubic splines, interpolated values of the dependent variable, and the definite integral of the dependent variable, respectively (same as in CONTWV). SPLIND has been added to perform the derivatives required in the normalization described in Section 1.2 above.

2.10 Functions HYD, CONHYP, HYPER2, and CMPGAMM

These functions calculate the hydrogenic wavefunction large component, as in (7). HYD is the main function routine, CONHYP is the confluent hypergeometric function evaluator of Nardin, et al [6], HYPER2 is the asymptotic approximation as in (8), and CMPGAMM is a complex gamma function.

3 Description of test data

The format for the test data is nearly the same for CONTWVG as it was for CONTWV but will be repeated for completeness. The test case was chosen on the basis of the fact that relativistic effects increase with atomic number, Z , and consequently mercury is a good candidate. One card is required to initiate the calculation of a continuum orbital: a card with the format of a label CONT, left justified and upper case, followed by a space, followed by the energy of the continuum orbital (referred to as variable ECONT by in CONTWVG), followed by a space, followed by the grid index point where the grid changes over from exponential to linear (referred to as variable MHALF in CONTWVG), followed by a space, followed by the file identification number where the continuum orbital is to be written, followed by a space, followed by "0" or "1" (referred to as IPHASE) for the phase shift calculation. If MHALF is zero (as in the TEST RUN INPUT), CONTWVG will automatically calculate an appropriate value for the parameter HCONT, the spacing for the linear portion of the two-piece radial grid. If IPHASE = 0, then the phase shift is not to be determined; if IPHASE = 1, then the phase shift is to be calculated.

This "CONT" card must follow the card which specifies the calculation mode (e.g. EAL, OL, etc.), and is the 37th line in the TEST RUN OUTPUT.

It should be noted that the principal quantum number referred to in the 25th line of the TEST RUN OUTPUT (-1 in this case) is only a dummy variable and is insignificant for the calculation.

Acknowledgments

The authors gratefully acknowledge the advice and encouragement given by Profs. Don R. Beck and James Waber, Michigan Tech University.

Adaptation Notes

The following steps must be performed to adapt the program of GRASP program to work with the continuum wavefunction solver, CONTWVG:

1) Delete subroutines DATAIN and SCF from the GRASP code.

2) Add CONTWVG (this code).

3) In subroutine MCDF, after the line:

```
: /SEMI/CHK({NC},{OL}),CCR({NC}, {OL})
```

add the line:

```
COMMON/GRIDC/RGRIDC({NPC}),ECONT,HCONT,MHALF, NCONT,NUMWR,IPHASE
```

4) In subroutine MCDF, after the line:

```
CALL INIT
```

add the four lines:

```
IF (ECONT.GT.ZERO) THEN  
  CALL MCPIN  
  GO TO 4  
END IF
```

5) In subroutine MCDF, change the line:

```
DO 6 NIT=1,NITIT
```

to

```
4 DO 6 NIT=1,NITIT
```

6) In subroutine MCDF, after the first occurrence of the line:

```
CALL SCF (ACCY,MX,DOALL)
```

add the line:

```
IF (ECONT.GT.ZERO) RETURN
```

7) Add the variable $NPC = 2000$ to the SYMBOL.DATA file, where 2000 is the number of points to use for the continuum wavefunction radial grid (use a larger number if higher energies are required).

It should be noted that CONTWVG must be preprocessed, with the same preprocessor as is used to preprocess the GRASP code (but with the variable NPC added to the SYMBOL.DATA file as described in the previous paragraph; then by including CONTWVG with the rest of the original GRASP program, it will automatically be preprocessed).

References

- [1] K. Dylla, I. Grant, C. Johnson, F. Parpia, and E. Plummer, *Comput. Phys. Commun.* **55** (1989) 425.
- [2] W. F. Perger and V. Karighattam, *Comput. Phys. Commun.* **66** (1991) 392.
- [3] W. Ong and A. Russek, *Phys. Rev. A.* **17** (1978) 120.
- [4] M. E. Rose, *Relativistic Electron Theory*, J. Wiley and Sons, New York, 1961.
- [5] W. F. Perger, Z. Cai, and D. R. Beck, *J. Phys. B: At. Mol. Opt. Phys.* **24** (1991) 4863.
- [6] M. Nardin, W. F. Perger, and A. Bhalla, *ACM Trans. Math. Softw.* **18** (1992) 345.
- [7] M. Nardin, W. F. Perger, and A. Bhalla, *J. Comput. Appl. Math.* **39** (1992) 193.

TEST RUN INPUT

MERCURY CONTINUUM WAVEFUNCTION RUN

1 23 0

1S

2S

2P-

2P

3S

3P-

3P

3D-

3D

4S

4P-

4P

4D-

4D

4F-

4F

5S

5P-

5P

5D-

5D

6S 1

-1S 1

ANG 1

1 1

MCP 8 8

MCDF 11

8 0 10 13 0

80.0 200.59

NUCLEUS FERMI

FIX 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22

LOAD 1 2

GRID RNT 1.0E-5 H 0.05

EAL 1

CONT 00.0100 0 14 1

END

TEST RUN OUTPUT