

Digital Computer Laboratory  
Massachusetts Institute of Technology  
Cambridge 39, Massachusetts

SUBJECT: PROGRAMS FOR SOLVING SECULAR EQUATIONS

To: Scientific and Engineering Computation Group

From: F.J. Corbató

Introduction

Production programs for Whirlwind I are now available for solving two types of secular equations by use of the techniques described in a report by A. Meckler<sup>1</sup> and are summarized briefly below. These are the ordinary variety, (case I),  $\sum_j H_{ij} \psi_{jk} = \psi_{ik} \lambda_k$ , and the general variety, (case II),  $\sum_j H_{ij} \psi_{jk} = \sum_j S_{ij} \psi_{jk} \lambda_k$ , where  $k = 1, 2, \dots, n$ ,

H and S are real symmetric matrices and S is positive definite. The two programs, for case I and for case II, can handle matrices of order  $1 \leq n \leq 32$ . The results are given photographically and consist of the input data and the  $\lambda_k$  and  $\psi_{jk}$  where in case I,  $\sum_j \psi_{ij}^\dagger \psi_{jk} = \delta_{ik}$

and in case II,  $\sum_{i,j} \psi_{li}^\dagger S_{ij} \psi_{jk} = \delta_{lk}$ . In addition the intermediate results of case II, if desired may be displayed.

Methods of Solution

Matrix diagonalization is the elementary process used to solve both cases. The diagonalization of a symmetric matrix  $M_{lk} = M_{kl}$  is accomplished by successive 2 by 2 "rotations" of all the matrix elements associated with the indices i and j where  $M_{ij}$  is the largest off-diagonal element. It can be simply shown that such a process converges. The explicit transformations applied are:

---

<sup>1</sup>A. Meckler, Quarterly Progress Report, Solid-State and Molecular Theory Group, M.I.T., Oct. 15, 1954, p. 15.

$$M'_{lk} = M_{lk} \quad l \neq i, j \quad k \neq i, j$$

$$M'_{ik} = cM_{ik} + sM_{jk} \quad k \neq i, j$$

$$M'_{jk} = -sM_{ik} + cM_{jk} \quad k \neq i, j$$

$$M'_{ii} = c^2 M_{ii} + 2scM_{ij} + s^2 M_{jj}$$

$$M'_{jj} = s^2 M_{ii} - 2scM_{ij} + c^2 M_{jj}$$

$$M'_{ij} = 0$$

where

$$c = \sqrt{\frac{1}{1+t^2}}$$

$$s = tc$$

and

$$t = \begin{cases} \frac{(M_{ii} - M_{jj}) + \sqrt{(M_{ii} - M_{jj})^2 + 4(M_{ij})^2}}{-2M_{ij}} & , (M_{ii} > M_{jj}) \\ \frac{+2M_{ij}}{(M_{ii} - M_{jj}) - \sqrt{(M_{ii} - M_{jj})^2 + 4(M_{ij})^2}} & , (M_{ii} \leq M_{jj}) \end{cases}$$

Similarly, the unitary transformation,  $U$ , (initially a diagonal unit matrix), is modified after each 2 by 2 rotation by the corresponding transformation affecting the  $i$ th and  $j$ th columns.

$$U'_{km} = U_{km} \quad m \neq i, j$$

$$U'_{ki} = cU_{ki} + sU_{kj}$$

$$U'_{kj} = -sU_{ki} + cU_{kj}$$

This process continues until  $M_{ij} < 2^c$ , where  $c$  is the preset criterion value, the result being that the diagonal of the final  $M_{ij}$  (i.e. the eigenvalues,  $\lambda$ ) and the final  $U$  satisfy the conditions:

$$\sum_{j=1}^W M_{ij} U_{jk} = U_{ik} \lambda_k \quad \text{or} \quad MU = U\lambda \quad (\text{where } \lambda \text{ does not multiply as a vector but as a scalar})$$

In nearly all applications, the original matrix,  $M$ , was computed from an operator,  $\hat{m}$ , by means of a set of basis states. These states can be represented as a set of unit vectors,  $\vec{e}_i$ , in a Hilbert space, and what is desired is the transformation which when applied to the original basis states gives a new set of basis states that when applied to the operator  $\hat{m}$ , yield a diagonal matrix. To obtain this transformation it is noted that if

$$M_{ij} = \langle \vec{e}_i | \hat{m} | \vec{e}_j \rangle$$

and

$$\delta_{lk} \lambda_k = \sum_{i,j} U_{li}^\dagger M_{ij} U_{jk}$$

then

$$\begin{aligned} \delta_{lk} \lambda_k &= \sum_{i,j} U_{li}^\dagger \langle \vec{e}_i | \hat{m} | \vec{e}_j \rangle U_{jk} \\ &= \left\langle \sum_i U_{li}^\dagger \vec{e}_i | \hat{m} | \sum_j U_{kj} \vec{e}_j \right\rangle \\ &= \langle \vec{e}_l' | \hat{m} | \vec{e}_k' \rangle \end{aligned}$$

Thus the desired transformation is  $\vec{e}_k' = \sum_j U_{kj}^\dagger \vec{e}_j$ . Stated in another

manner, the kth eigenvalue of  $M$  is associated with an eigenvector consisting of the kth column-vector of  $U_{jk}$  where the components of the eigenvector refer to the original basis states used to compute  $M$ .

For case II, the procedure of solution is the following. First  $S$  is diagonalized so that  $S U_{S'} = U_{S'} S'$ , where  $S'$  is diagonal. Then  $S^{-1/2}$  is formed where  $S^{-1/2} = U_{S'} (S')^{-1/2} U_{S'}^\dagger$ , and the new matrix  $H'$  is formed by matrix multiplication where  $H' = S^{-1/2} H S^{-1/2}$ . Then a second diagonalization is made such that

$$H' U_{H'} = U_{H'} \lambda$$

and finally  $\Psi = S^{-1/2} U_{H'}$ , where it is seen that  $\Psi^\dagger S \Psi = 1$ .

To verify that this  $\lambda$  and  $\Psi$  form desired solutions, it is observed that

$$\begin{aligned}
H\Psi &= HS^{-1/2}U_{H'} = \left[ U_{S'}(S')^{-1/2}U_{S'}^\dagger, U_{S'}(S')^{-1/2}U_{S'}^\dagger \right] HS^{-1/2}U_{H'} = \\
&= S^{1/2}H'U_{H'} = S^{1/2}U_{H'}\lambda = \\
&= \left[ U_{S'}(S')^{-1/2}U_{S'}^\dagger \right] \left[ U_{S'}(S')^{-1/2}U_{S'}^\dagger, U_{S'}(S')^{-1/2}U_{S'}^\dagger \right] U_{H'}\lambda \\
&= SS^{-1/2}U_{H'}\lambda \\
&= S\Psi\lambda .
\end{aligned}$$

#### Specification of Input Data

Both programs have their input data, (also called parameter sets), located in the same storage locations and moreover do not destroy during operation any of the necessary input information. Thus a series of secular equations each differing from the previous by the addition of a row and column could be conveniently solved by supplying the complete matrix (or pair of matrices in case II) in the first parameter tape and then merely changing the matrix order,  $n$ , for each subsequent parameter. Furthermore, a parameter tape for case II could be used with program I, but would, of course, give solutions for case I. (Program I would also disturb the storage of the unnecessary  $S$  matrix.)

In the listing of the input data, two types of number conventions are used. These are the single-register (15,0) integer, (less than 32768 in magnitude), which may have a sign but no decimal point, and the double-register (24,6) generalized decimal number which must have both a sign and a decimal point. The specific locations of the input data are:

34	+n	matrix order, $1 \leq n \leq 32$	} (15,0) integers
35	+c <sub>S</sub>	diagonalization criterion of $S$ , $c_S \geq -60$	
36	+c <sub>H</sub>	diagonalization criterion of $H$ , $c_H \geq -60$	
37	+k	identification number	
2048	+H <sub>11</sub>	listing of the $H$ matrix	}

(continued on next page)

	+H <sub>12</sub>		} (24,6) numbers
	+H <sub>22</sub>		
	+H <sub>13</sub>		
	+H <sub>23</sub>		
	....		
	+H <sub>nn</sub>		
3104	+S <sub>11</sub>	listing of the S matrix	
	+S <sub>12</sub>		
	+S <sub>22</sub>		
	+S <sub>13</sub>		
	+S <sub>23</sub>		
	....		
	+S <sub>nn</sub>		

Both programs start at register 32 decimal and stop on an "si 0" instruction in register 33 decimal (41 octal). If many sets of parameters are to be included in one big tape and are to be automatically run consecutively, all but the last set should contain an additional 33 | sp26, and the last set should contain a 33 | si0. Each set on a multiple-parameter tape must be preceded by a "fc" and then a "(24,6)"; the performance request should have, under the heading of operation instructions, a "Turn off si 1 switch" on the line preceding the multiple-parameter tape operation instruction.

### Criterion Values

The values of  $C_H$  and  $C_S$  are used to terminate the diagonalization procedures after sufficient accuracy has been obtained. For both the H and the S diagonalization, each c should be chosen such that

$$\frac{2^c}{|\lambda_i - \lambda_j|_{\max}} \leq \frac{\left| \frac{\text{maximum absolute error in eigenvector components}}{1} \right|}{\left| \frac{\text{maximum absolute error in eigenvalues}}{|\lambda_k|_{\max}} \right|} \approx$$

The accuracy of the computations is such that the right-hand side of the above relation can never be made less than about  $10^{-7}$ . Empirical values for this last figure are:

	Case I (n = 5)	Case II (n = 5)
Eigenvalues:	$1 \times 10^{-7}$	$4 \times 10^{-7}$
Eigenvectors:	$2 \times 10^{-7}$	$5 \times 10^{-7}$
	Case I (n = 18)	
Eigenvalues:	$3 \times 10^{-7}$	
Eigenvectors:	$8 \times 10^{-7}$	

For an example of the determination of  $c$ , suppose that 6 figures were designed in the eigenvectors and that  $|\lambda_i - \lambda_j|_{\max} \sim 3$ . Then

$$2^c \leq 3 \times \frac{\frac{1}{2} \times 10^{-7}}{1} = 150 \times 10^{-9} < 256 \times 10^{-9} \cong 2^8 2^{-30} = 2^{-24}$$

and therefore  $c_{\max} \cong -24$ . (A useful relation is that  $2^{10} = 1024 \cong 10^{+3}$ ). In practice, one would probably use  $+c = -30$  to be certain of the desired accuracy. As is implied, the value of  $c$  can always be safely lowered (except not below approximately  $-60$ ) since the only effect will be to raise the computation time somewhat (i.e. at worst up to 50 or 100% more time). For case II when the eigenvalues of  $S$  and  $H$  are both of the same order,  $c_S$  and  $c_H$  should be about the same, since the overall accuracy is determined by the least accurate diagonalization.

### Form of Output

All of the program output is given photographically with each secular equation solution beginning on a new frame. The various displays may be divided into three classes: 1) the initial data, 2) the intermediate results (case II only), and 3) the final results. Each display of the first two classes will consist of a "heading" line followed by either a "symmetric" or "square" pattern of the pertinent numbers. The third class consists of a heading line followed by the "rectangular" pattern of  $\lambda_k$  in the first column and then the  $\psi_{j1}$  to  $\psi_{jn}$ , each in a

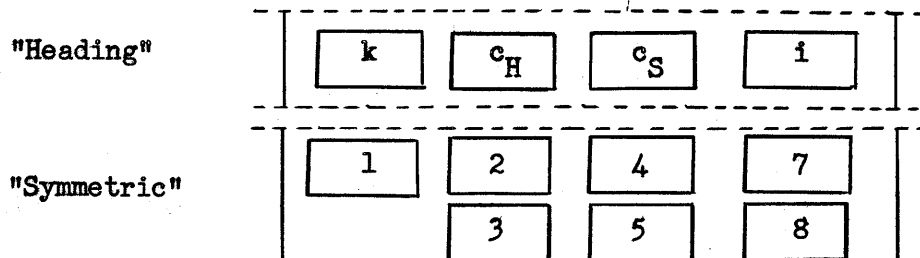
column, starting from the third column. The first class is normally displayed but may be suppressed, the second class is normally suppressed but may be displayed, and the third class is always displayed. The program will use as many frames as are necessary to include all the displays, starting on a new frame if a particular display will not entirely fit on the remainder of a frame.

The "heading" line will have in the first column, the identifying code number of the solution,  $k$ ; in the second column the criterion value,  $c_H$  (except on the display of  $S$ ); in the third column the criterion value  $c_S$ , (except on the display of  $H$ ); and in the fourth column, an intermediate result identification number,  $i$ , (only for class 2 displays). Case I results may be distinguished from Case II results by the absence of  $c_S$  in the result heading line.

The selection or suppression of the various displays depends on the contents of the associated suppressor registers: +0 for display; -0 for suppression. These displays and their suppressor registers are:

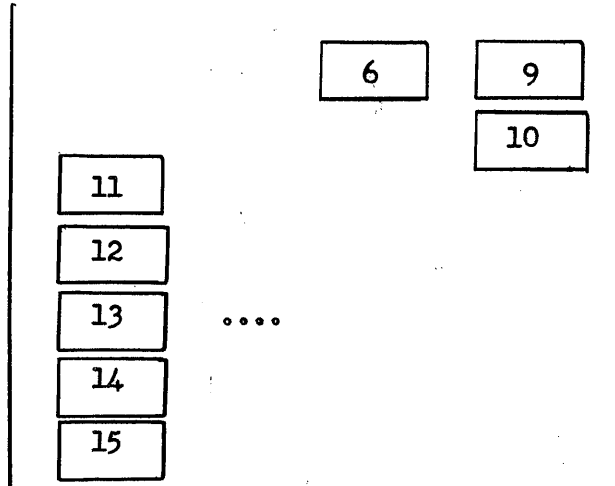
	Display	Contents	Pattern	Decimal Address of Suppressor Register
Class 1	{ H	H	sym.	1598
	{ S	S	sym.	
Class 2	{ 1	$S'$	sym.	1599
	{ 2	$U_{S'}$	sq.	1600
	{ 3	$S^{-1/2}$	sym.	1601
	{ 4	$HS^{-1/2}$	sq.	1602
	{ 5	$H'$	sym.	1603
	{ 6	$U_{H'}$	sq.	1604
Class 3	{ $\lambda + \psi$	$\lambda + \psi$	rect.	—

Symbolically, we have for the various forms of output:

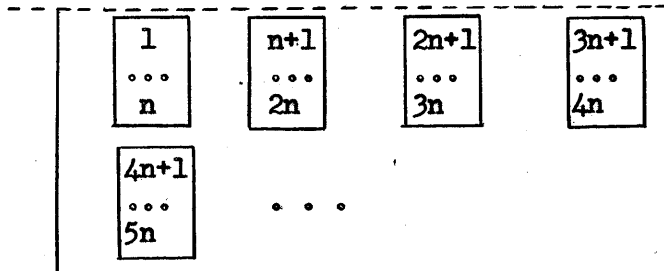


(continued on next page)

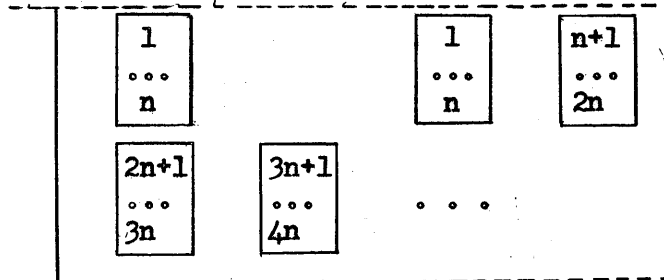
"Symmetric"  
(continued)



"Square"



"Rectangular"



The total number of film frames required per solution depends on the number of displays not suppressed but may be computed from the schedules given below. If two or more displays are to fit on the same frame, they will be separated by two additional spacing lines beyond that given in the schedules. (For  $n > 8$ , no two displays will fit on the same frame.) Each frame can contain a maximum of 36 lines.

matrix order	number of lines (or frames = f)		
	head. + sym.	head. + sq.	head. + rect.
1	3	3	3
2	4	4	4
3	5	5	9

(continued on next page)



4	6	6	11
5	12	13	13
6	13	15	15
7	14	17	25
8	15	19	28
9-10	f	f	f
11-12	f	2f	2f
13-14	2f	2f	2f
15-16	2f	2f	3f
17	2f	3f	3f
18	2f	5f	5f
19	2f	5f	6f
20	3f	5f	6f
21-22	4f	6f	6f
23-24	4f	6f	7f
25-26	5f	7f	7f
27-28	5f	7f	8f
29-30	6f	8f	8f
31-32	6f	8f	9f

For convenience the most common situations are summarized:

matrix order	number of frames per solution		
	normal case I	normal case II	full suppression of cases I and II
1-4	1	1	1
5-6	1	2	1
7-8	2	2	1
9-10	2	3	1
11-12	3	4	2
13-14	4	6	2
15-17	5	7	3
18	7	9	5
19	8	10	6
20	9	12	6

(continued on next page)

21-22	10	14	6
23-24	11	15	7
25-26	12	17	7
27-28	13	18	8
29-30	14	20	8
31-32	15	21	9

### Information for Filling Out Performance Requests

1. Time estimates: These are difficult to state in general but roughly  $T_{\text{case II}} \cong 2T_{\text{case I}}$  and  $T_{\text{case I}} \cong Kn^3$  where  $K \sim \frac{1}{10}$  sec. Variations from this formula depend on the difficulty of diagonalization (i.e. on the size of the off-diagonal elements and the stringency of the criteria). \*

2. Program stop: Stops automatically on si0 in register 41 octal.

3. Camera output: See film frame schedules above for the number of frames per solution. Camera first used in 0 sec.

4. Magnetic drum: Case I uses auxiliary drum groups 1, 2, 3; case II uses auxiliary drum groups 1, 2, 3, 4. Drum first used in 0 sec.

5. Operating instructions: These are for a single-parameter tape

E, fb A , RI,  
fb 10172-20-B, RI,  
fc C , RI, RS.

where A = "dummy logging tape";  
(tape room will prepare  
this)

B =  $\begin{cases} 334, & \text{(case I)} \\ 331, & \text{(case II)} \end{cases}$

C = parameter tape number

For multiple-parameter tapes, the corresponding instructions are

E, fb A , RI,  
fb ~~10172-20-B~~<sup>10172-20-B</sup>, RI,  
Turn off si 1 switch  
fc C , RI.

\* An additional time of about 10 seconds must be added for each full frame displayed.

Program Alarms

<u>Type of Alarm</u>	<u>Octal Identification</u>	<u>Immediate Cause</u>	<u>Probable Cause</u>
1. Check register	PC = 3764 AR = 3453	GD no. unscale- factored	a
2. Check register	PC = 3764 AR = 3503	GD no. in MRA too large to store	a,b
3. Check register (case II only)	PC = 2601	An eigenvalue of S is < 0	a,b,c
4. Divide error (case II only)	PC = 3612	An eigenvalue of S = 0	a,b,c

Probable Causes

- a. GD number of initial data out of phase, due to either a missing decimal point, a missing seventh hole in the tape, an incorrect storage address, or a faulty parameter tape read-in by the computer. Compare print of data tape with display of input data.
- b. Input data is very poorly scale-factored or, (case II only), one of the eigenvalues of S is extremely small.
- c. The matrix S is not positive definite.

Additional Remarks for Experienced Programmers

1. If matrix elements are to be generated, decimal registers 38 to 1563 inclusive are available for programs and will be restored after each solution. Additional storage is available on the auxiliary drum starting at the decimal address 7630 (case I) and 10206 (case II). All generation program tapes must have a NOT PA included; the PA already in the program contains buffers b and 6b inclusive and a single cycle counter. Thus all cycle instructions may be used, but no "isc" orders (except isc0). Buffers b through 3b are used as temporary storage during a solution and 4b through 6b are unused.

2. The program also contains a DIB/DOB subroutine; the decimal entries are:

for DIB

for DOB

sp1605

sp1610

[CM] core memory address

[CM]

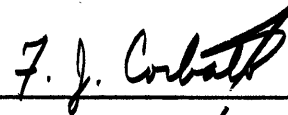
[DA] aux. drum address

[DA]

w no. of reg. to transfer

w

← return point in wi mode →



---

F.J. Corbató

March 15, 1955