Hydrological Sciences Journal

ISSN: 0262-6667 (Print) 2150-3435 (Online) Journal homepage: https://www.tandfonline.com/loi/thsj20

Steady state groundwater model for a programmable calculator

ROBERT G. THOMAS

Hydrological Sciences – Journal – des Sciences Hydrologiques, 27, 1, 3/1982

Steady state groundwater model for a programmable calculator

ROBERT G. THOMAS
Senior Officer, Water Resources, Development and Management Service, Land and Water Development Division, Food and Agriculture Organization of the United Nations, Rome, Italy

ABSTRACT The steady state square node groundwater model presented is for a confined aquifer with choice of recharge/discharge and constant head boundaries. Transmissivity, recharge/discharge, constant and initial head and nodal network must be input. Output is steady state water levels to the precision desired by the operator up to calculator limit of significant figures. Forty nodes or less can be defined but depending on complexity the program can be easily modified for variable thickness and floating fresh water lens conditions.

Modèle d'écoulement souterrain en régime permanent pour calculateur de poche programmable
RESUME Le modèle d'écoulement souterrain à mailles carrées présenté s'applique à une formation aquifère captive en régime permanent. Les conditions aux limites (débits ou charges constantes), la transmissivité, la recharge/extraction, les charges piézométriques initiales et la géométrie du réseau sont donnés. Les résultats sont les niveaux d'eau en régime permanent, calculés avec la précision désirée jusqu'à la limite de la capacité de la calculatrice. Un maximum de 40 noeuds peut être défini, et selon la complexité du problème, le programme peut être facilement adapté pour un aquifère d'épaisseur variable ou des conditions de lentille d'eau douce flottant sur de l'eau salée.

INTRODUCTION

A very large number of mathematical models for solution of a wide range of groundwater flow conditions have recently been summarized by Bachmat *et al.* (1980). These include steady state models much more sophisticated than the one described here. However, most groundwater models have been run on computers of larger or smaller size. Most of these require programming expertise and hardware which is not easily portable. Some of the newer minicomputers are nearly portable and perhaps soon they will be easily available. For several years programmable pocket calculators of several makes and with a variety of accessories have been available, which are truly portable. It is not the purpose of this paper to compare the various brands of programmable calculator nor to say which is best since this is strictly a personal choice and involves a wide range of factors other than portability.

The steady state groundwater model outlined here has been programmed on the TI-59 but can also be programmed on similar calculators of other brands. The writer finds that the availability of a printer greatly reduces debugging time and increases the usefulness of the calculator. However, the same groundwater model can be used if a printer is not available. For example, the program described here repeats itself endlessly, printing results at the end of each iteration. The user stops the calculator when sufficiently precise results are obtained. Sometimes for rough work only three or four significant figures are needed. For more precise results the program can continue until the significant figures on the calculator are reached (8 or 10). However, program modifications can include use with no printer, and automatic stopping when a particular precision is obtained.

The difficulty with most programmable calculators is the relatively limited memory and program storage capacity. Therefore few people try to solve complex groundwater models with them. However, steady state models are relatively simple mathematically, particularly the one used here. If water level results are printed or read after each iteration or each few iterations it is not necessary to store more than one set of water level data making the solution practical for programmable calculators.

The program presented in Table 1 calculates the steady state for a square node, two dimensional, confined groundwater model. The number of nodes available depends on the number of transmissivity and input/output values that are desired. The printer is essential for easy data output. It takes about 15 s for one basic water level calculation and therefore 1 min for one iteration of a four node problem.

For a 30 node problem it may take several hours for precision to eight significant figures if the problem is complex. With the printer the results of each step are visible and the program is stopped manually when sufficient precision is obtained. The print instruction can be deleted to save paper and the heads listed when the program is stopped. Much time is saved if the initial heads used are reasonably close to the likely solution.

THE MODEL

The model is for a single aquifer. Transmissivity (T) and input (+)/ extraction (-) (or AQ) can be distributed among the nodes as desired. T is assumed to be for the node. The nodes are square and all of the same size. The squares can be arranged in any manner convenient to the problem. As many constant head values as desired are provided for but take up more storage and decrease the possible number of active nodes.

The unit of transmissivity (T) must coincide with the unit for input/extraction (AQ), that is, the unit for T must equal the unit for AQ divided by whichever length unit is being used (e.g. T in m^2 $year^{-1}$, AQ in m^3 $year^{-1}$). With a total of eight AQ and T values

Table 1 Program for the steady state groundwater model

Loc	Code	Key	Comments	Loc	Code	Key	Comments	Loc	Code	Key	Comments
000	01	1	Initial	055	85	+	Find	110	96	96	
001	42	STO	iteration	056	43	RCL	loc of	111	99	PRT	Print H(N)
002	90	90	count	057	98	98	T_i and	112	01	1	
003	76	LBL		058	95	=	store	113	44	SUM	Add 1
004	13	C		059	42	STO		114	96	96	to
005	43	RCL		060	93	93		115	44	SUM	counters
006	00	00		061	73	RC*		116	91	91	
007	42	STO		062	93	93		117	97	DSZ*	If all
008	87	87	Determine	063	59	INT	Find T_i	118	87	87	H(N) calc.
009	43	RCL	and store	064	42	STO		119	14	D	go to D
010	98	98	coefficients	065	93	93		120	98	ADV	
011	85	+	and	066	73	RC*	T_i	121	01	1	
012	01	1	memory	067	93	93		122	44	SUM	
013	95	=	locations	068	85	+	+	123	90	90	
014	42	STO		069	73	RC*		124	43	RCL	
015	91	91		070	92	92	T_N	125	97	97	Print
016	01	1		071	95	=		126	69	OP	iteration
017	42	STO		072	55	÷	÷	127	04	04	number
018	96	96		073	02	2	2	128	43	RCL	
019	76	LBL		074	95	=	$= \frac{T_i + T_N}{2}$	129	90	90	
020	14	D		075	44	SUM		130	69	OP	
021	00	0	Store	076	89	89	Σ	131	06	06	
022	42	STO	zero in	077	65	X	X	132	61	GTO	
023	88	88	RES	078	53	(		133	13	C	
024	42	STO	and	079	73	RC*		134	76	LBL	
025	89	89	RELAX	080	95	95		135	89		
026	73	RC*	Store	081	75	−	$(H_i - H_N)$	136	01	1	
027	91	91	NI(N) in	082	73	RC*		137	52	EE	Subroutine
028	42	STO	working	083	96	96		138	02	2	to
029	85	85	memory	084	54	)		139	49	PRD	extract
030	59	INT		085	95	=	=	140	85	85	AQ and
031	42	STO	Find TN	086	44	SUM	ΣQ_{iN}	141	43	RCL	H
032	92	92	location	087	88	88		142	85	85	locations
033	22	INV	and modify	088	97	DSZ*	Counter	143	59	INT	
034	44	SUM	for AQ	089	86	86	for loop	144	22	INV	
035	85	85		090	11	A		145	44	SUM	
036	71	SBR		091	76	LBL		146	85	85	
037	89			092	12	B		147	92	RTN	
038	42	STO	Store AQ_N	093	73	RC*					
039	94	94	location	094	94	94	AQ				
040	04	4	Counter	095	85	+	+				
041	42	STO	for	096	43	RCL	ΣQ_{iN}				
042	86	86	loop	097	88	88					
043	76	LBL		098	95	=					
044	11	A		099	55	÷	—				
045	71	SBR		100	43	RCL	RELAX				
046	89			101	89	89					
047	42	STO	Store H_i	102	65	X	X				
048	95	95	location	103	43	RCL	COEF				
049	32	X=T		104	99	99					
050	00	0		105	85	+	+				
051	67	EQ		106	73	RC*	H_{old}				
052	12	B		107	96	96					
053	43	RCL		108	95	=	=				
054	95	95		109	72	ST*	H_{new}				

Special entry for DSZ.

and two constant head nodes there is a limit of 36 active nodes. With one constant head node, one T value and one AQ value up to 40 nodes can be used with the TI-59.

THE ALGORITHM

The algorithm used is a simplified version of the Tyson-Weber (1964) solution which Narasimhan & Witherspoon (1976) call an integrated finite difference method. It is likely that other kinds of numerical solutions will be found which converge more rapidly than this one. Speed of solution on programmable calculators is a matter of personal choice. The programming to speed up the calculation sometimes takes longer than the calculation itself. The main reason the writer chose this algorithm over others was simply his familiarity with it. The reader who is interested and feels happier with a finite difference solution will no doubt be able to use that algorithm.

The main aspect of this paper is not so much the algorithm (or the brand of calculator) but the way to store as much information as possible in as little space as possible.

Indirect addressing and twelve digit internal memory capability are used for this purpose and should be available on the calculator.

The program calculates the flow Q_i across nodal interfaces. Thus:

$$Q_i = \left[(T_i + T_N)/2\right] (H_i - H_N) \qquad (1)$$

where T_i and T_N are transmissivities of the adjacent node, i, and the node under consideration, N.

The value $(T_i + T_N)/2$ is saved and accumulated for use as RELAX.

$$RELAX_N = \Sigma_1^4 \left[(T_i + T_N)/2\right] \qquad (2)$$

The value RES for node N is also accumulated as RES_N.

$$RES_N = \Sigma_1^4 Q_i + AQ_N \qquad (3)$$

The new head H is obtained by:

$$H_{new} = H_{old} + (RES_N{*}COEF)/RELAX_N \qquad (4)$$

COEF is a relaxation coefficient which can be varied, but 1.4 is generally best. If COEF is too small the solution converges very slowly. If it is too large the solution converges slowly in the manner of a Fourier series curve.

The program calculates a new head for each node and starts a new calculation using the previously calculated heads. One iteration consists of one complete set of new head calculations for all nodes, which are printed. Each time a new head is calculated it replaces the old head and is used immediately.

The elements of arithmetic are quite simple as seen above. The problem with the limited memory calculators is to store nodal identification and other data.

On the TI-59 the automatic decrementing function, DSZ, is used. This is followed by the memory location to be decremented (say DSZ0). On the TI-59 only a one digit number can be directly entered which restricts use to the first 10 memories and would cause complications in head location numbering. However, a two digit number will work once entered. An example of entry for DSZ99 follows: DSZ, RCL, 99, BST, BST, DEL, SST.

DATA STORAGE ALLOCATION

From this point on the discussion generally will refer to the TI-59 calculator. Users of other calculators should be able to prepare their own program on this basis. Table 1 lists the program and Table 2 gives instructions for its use.

Table 2 How to use the programme for the steady state groundwater model

Step	Procedure	Enter	Press	Output/Mode
1	Prepare the nodal layout on a sufficiently large paper to use as a worksheet. Assign T values to all nodes and AQ values for any nodes of interest.			
2	Prepare a list of input values from the map as shown by the example.			
3	Memory partition.	10	2nd Op 17	159.99
4	Clear old programs and data, load the groundwater program by hand* or by magnetic card if available. List the program and the memory storage to make sure of the contents.			
5	Enter NN.	NN	STO 00	NN
6	Enter COEF.	1.4	STO 99	1.4
7	Enter NNT.	NNT	STO 98	NNT
8	Enter print number.	64243735	STO 97	64243735
9	Starting at memory 01 enter the approximation of the head values ending with the constant heads if needed. The initial head values could be zero but an unnecessary number of iterations are then required and it is better to make a rough estimate for initial conditions.			
10	Immediately after the constant heads, store the 12 digit nodal data numbers, ending with the constant head nodes in the same order as the constant water levels were entered. List the memory storage to make sure the proper entries have been made.			
11	To start the program press R/S.			
12	To stop the program press R/S.			
13	To start with new data press RST and R/S, being sure to enter appropriate initial head estimates and to clear memories 85-96 to zero.			

* If entered by hand special care is needed to enter DSZ since memories used exceed 9. Also pay special attention on entering NI values since 12 numbers may be needed.

The data storage system is divided into five main categories (a) initial water level data; (b) location of data in storage; (c) T, AQ, or other input data; (d) memory needed for internal calculation; and (e) program constants. Note that less than 159 program steps are used and therefore 99 memory allocations are made available. The operation 10 2nd Op 17 must be done before reading from a magnetic card or before storing the instructions manually. If the program is modified to have more than 159 steps less space is available for memory storage.

Initial water level data For each node, space is needed for the head value and for a number representing location of data information. In this program the head values are stored starting in memory 01. The number of active nodes are NN and the active nodes plus the constant head nodes are NNT. Thus active node initial heads and calculated heads are stored in M(01) to M(NN) and the constant heads in M(NN+1) to M(NNT). The identification of the nodes is by the number where the head information is stored. Thus node 1 head value is stored in memory 01, etc.

Location of data in storage For each active node three kinds of information are needed. These are the T of the node, AQ value, and identity of up to four adjacent nodes. This means that 12 two-digit numbers are required. The face of the TI-59 and the printer only holds 10 digits but the memory chip holds two more which can be used. The scheme for storing the 12 digits and their meaning follows and is needed for each active node.

	decimal point		
X X	•	X X	XX XX XX XX
T location		AQ location	two digit numbers showing location of heads for up to four adjacent nodes

For the constant head nodes, only the T value location is needed and other data need not be entered at all. Note that if there are less than four adjacent nodes the number can be entered directly, thus X X . X X X X X X 0 0 ... since the last two hidden digits will be zero.

For a full 12 digit number first enter . X X X X X X X X X X (the decimal point and the last 10 digits) and then using SUM *add* the *first two* digits to the same memory. The subroutine takes the two digit numbers out one by one after the first two are obtained by using the INTEGER function. These node information numbers are stored in M(NNT+1) to M(2NNT).

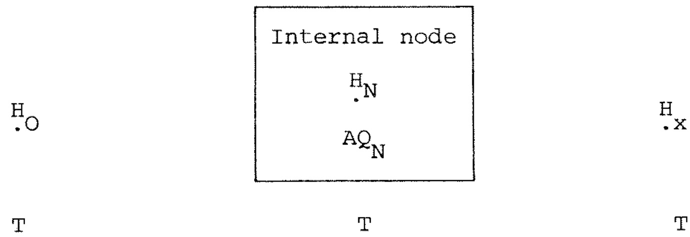
T, AQ or other input data Transmissivity (T) and input/output (AQ) values for the confined aquifer are assigned arbitrary numbers following the location data. In other words these start at 2NNT + 1 and can continue through memory 84. In other variations of the program these spaces can be used for permeability, base of aquifer, etc.

Memory needed for internal calculations Memory locations 85 to 96 are used for internal calculations and need to be set to zero when a program is started from initial conditions.

Program constants Program constants include one value for the printer and various constants depending on the model.

Memory location	*Enter*
00	NN
99	Relaxation Coefficient
98	NNT
97	64243735

To illustrate the data storage system, a single internal node with two constant head nodes is shown below.



Store T in 07, store AQ_N in 08.

Head storage location is 03 for H_O, 01 for H_N and 02 for H_x.

Then the node information (NI) is 7.080302000O. There must be 12 numbers if all four interconnects are used. The first number in front of the decimal point is the location of T, the next two digits, 08, is location of AQ, the next two digits 03, give location of head value for H_O and 02 gives head value location for H_x.

This number is stored in the next available memory starting at (NNT + 1). Thus:

Mem	N_i
04	7.080302
05	7.0
06	7.0

Note that the T location for the constant head nodes must be entered in the same order as the constant head locations were entered. Inspection of the example in Table 3 will show how this works.

MODIFICATIONS

The writer has modified this program for a confined horizontal base aquifer, an unconfined variable base and for an unconfined floating fresh water lens. The only change in the algorithm is for T_i as shown in Table 4. The writer arbitrarily stored the constant used (B, P, or U) in mem 84 but mem 97 can be used if "ITER" is deleted. The location of values PN, or B_N, is handled just as T_N for the confined aquifer. The changing of the confined aquifer program is not difficult once the memory allocations are understood. The definitions of the data are shown in Table 5.

Since a considerable amount of paper is used the print statement (Step 111) can be deleted. The head values can then be listed after pressing R/S to stop the calculation. After printing the head values the program can be started again provided care has been taken to have the same number on the face of the calculator. It is best therefore to stop the calculation when 0 or 1 is shown on the calculator face.

The statements for printing the iteration number can be simplified in order to be able to keep the modified program within 159 steps.

Table 3 Four node example

Head number and location	$H_0 = H_5$	H_1	H_2	H_3	H_4	$H_x = H_6$
T values and memory locations						
T value	10 000	5000	5000	5000	4000	1000
Location of T	14	15	15	15	16	17
AQ values and memory locations						
AQ value		0	0	0	0	
Location of AQ		18	18	18	18	
H value and locations						
Initial H value	100	90	70	30	10	0
Location of H	05	01	02	03	04	06

Number of active nodes	4	mem 00
Relaxation coefficient	1.4	mem 99
Total number of nodes	6	mem 98

Nodal data starts at memory 01 in this example and continues through 6 + 6 = 12.

The listing below shows the initial data for this problem:

4.	00
90.	01
70.	02
30.	03
10.	04
100.	05
0.	06
15. 180502	07
15. 180103	08
15. 180204	09
19. 180306	10
14.	11
17.	12
0.	13
10000.	14
5000.	15
4000.	16
1000.	17
0.	18
0.	19
0.	20
0.	21
0.	22
2.	95
1.	96
64263735.	97
6.	98
1.4	99

The listing below shows the early and final results of the calculation:

8.72	01	
5.404	01	
4.3104	01	
4.6288	01	
2.	00	ITR
7.93824	01	
6.412448	01	
6.4208981	01	
5.6395278	01	
3.2	01	ITR
9.5454545	01	
8.8636384	01	
8.1818182	01	
6.8181818	01	
3.3	01	ITR
9.5454545	01	
8.8636384	01	
8.1818182	01	
6.8181818	01	

Values for T, AQ, etc. can be changed as the program progresses but this usually is not of much value until a suitable precision has been reached.

Table 4 Modifications to the program

Condition	Equation for T	Other variables
Confined aquifer	$T_i = \frac{T_i + T_N}{2}$	
Unconfined – flat base variable permeability	$T_i = \frac{(H_N - B)PN + (H_i - B)P_i}{2}$	B = elevation of base of aquifer
Unconfined – variable base constant P	$T_i = \frac{(H_N - B_N)P + (H_i - B_i)P}{2}$	P = permeability
Floating lens	$Ti = \frac{P_N H_N U + P_i H_i U}{2}$	$U = 1 + \gamma_F/(\gamma_s - \gamma_F)$ γ_F = density of fresh water γ_s = density of salt water

Table 5 Data registers

00	NN	92	T(N) Location
01 to 83	HN, NI, T, AQ	93	T(I) Location
85	Working Store	94	AQ(N) Location
86	Interconnect Index	95	H(I) Location
87	Node Index	96	H(N) Location
88	RES	97	64243735
89	RELAX	98	NNT
90	ITER NO	99	COEF
91	NI Location		

REFERENCES

Bachmat, Y., Bredehoeft, J., Andrews, B., Holy, D. & Sebastian, S. (1980) *Groundwater Management: The Use of Numerical Models.* AGU, Water Resources Monograph 5.

Narasimhan, T.N. & Witherspoon, P.A. (1976) An integrated finite difference method for analyzing fluid flow in porous media. *Wat. Resour. Res.* 12(1), 57-64.

Tyson, H.N. & Weber, E.M. (1964) Groundwater management for the nations future - Computer simulation of groundwater basins. *J. Hydraul. Div. ASCE* 90 (HY4), 59-77.

Received 23 February 1981; accepted 30 December 1981.