Chapter 7
Solving multiple equations

Many problems of science and engineering require the simultaneous solutions of more than one equation. The calculator provides several procedures for solving multiple equations as presented below. Please notice that no discussion of solving systems of linear equations is presented in this chapter. Linear systems solutions will be discussed in detail in subsequent chapters on matrices and linear algebra.

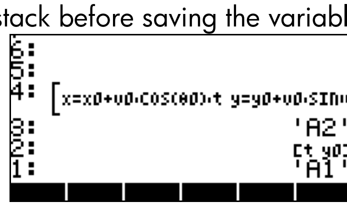
Rational equation systems

Equations that can be re-written as polynomials or rational algebraic expressions can be solved directly by the calculator by using the function SOLVE. You need to provide the list of equations as elements of a vector. The list of variables to solve for must also be provided as a vector. Make sure that the CAS is set to mode Exact before attempting a solution using this procedure. Also, the more complicated the expressions, the longer the CAS takes in solving a particular system of equations. Examples of this application follow:

Example 1 – Projectile motion

Use function SOLVE with the following vector arguments, the first being the list of equations: ['x = x0 + v0*COS(θ0)*t' 'y =y0+v0*SIN(θ0)*t – g*t^2/2'] ENTER, and the second being the variables to solve for, say t and y0, i.e., ['t' 'y0'].

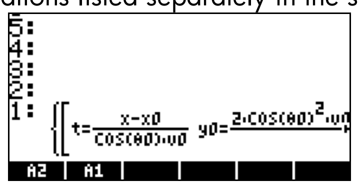
The solution in this case will be provided using the RPN mode. The only reason being that we can build the solution step by step. The solution in the ALG mode is very similar. First, we store the first vector (equations) into variable A2, and the vector of variables into variable A1. The following screen shows the RPN stack before saving the variables.

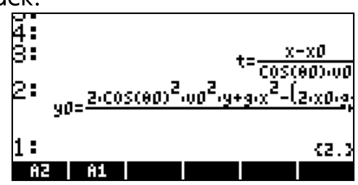


At this point, we need only press [STO▸] twice to store these variables.
To solve, first change CAS mode to Exact, then, list the contents of A2 and A1, in that order: [A2] [A1] .

Use command SOLVE at this point (from the S.SLV menu: [←] S.SLV) After about 40 seconds, maybe more, you get as result a list:
{ 't = (x-x0)/(COS(θ0)*v0)'
'y0 = (2*COS(θ0)^2*v0^2*y+(g*x^2(2*x0*g+2*SIN(•0))*COS(θ0)*v0^2)*x+
(x0^2*g+2*SIN(θ0)*COS(θ0)*v0^2*x0)))/(2*COS(θ0)^2*v0^2)']}

Press [EVAL] to remove the vector from the list, then use command OBJ→, to get the equations listed separately in the stack.





Note: This method worked fine in this example because the unknowns t and y0 were algebraic terms in the equations. This method would not work for solving for θ0, since θ0 belongs to a transcendental term.

Example 2 – Stresses in a thick wall cylinder

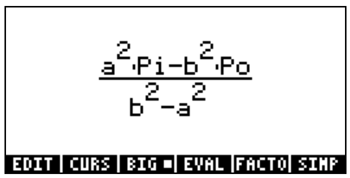
Consider a thick-wall cylinder for inner and outer radius *a* and *b*, respectively, subject to an inner pressure P_i and outer pressure P_o. At any radial distance *r* from the cylinder's axis the normal stresses in the radial and transverse directions, σ_{rr} and $\sigma_{\theta\theta}$, respectively, are given by

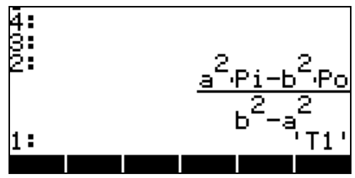
$$\sigma_{\theta\theta} = \frac{a^2 \cdot P_i - b^2 \cdot P_o}{b^2 - a^2} + \frac{a^2 \cdot b^2 \cdot (P_i - P_o)}{r^2 \cdot (b^2 - a^2)},$$

$$\sigma_{rr} = \frac{a^2 \cdot P_i - b^2 \cdot P_o}{b^2 - a^2} - \frac{a^2 \cdot b^2 \cdot (P_i - P_o)}{r^2 \cdot (b^2 - a^2)}.$$

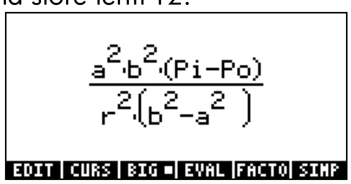
Notice that the right-hand sides of the two equations differ only in the sign between the two terms. Therefore, to write these equations in the calculator, I suggest you type the first term and store in a variable T1, then the second term, and store it in T2. Writing the equations afterwards will be matter of recalling the contents of T1 and T2 to the stack and adding and subtracting them. Here is how to do it with the equation writer:

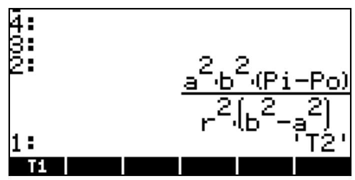
Enter and store term T1:





Enter and store term T2:

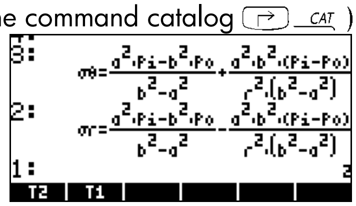


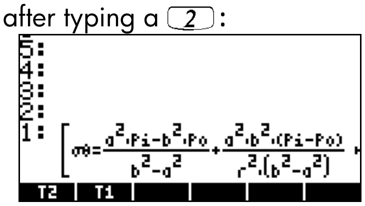


Notice that we are using the RPN mode in this example, however, the procedure in the ALG mode should be very similar. Create the equation for $\sigma_{\theta\theta}$: [VAR] [T1] [T2] [+] [ALPHA][→][S] [ALPHA][→][T] [ENTER] [▶] [→] =

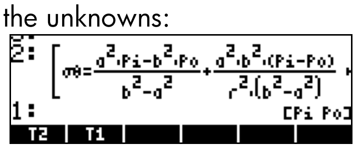
Create the equation for σ_{rr}: [VAR] [T1] [T2] [−] [ALPHA][→][S] [ALPHA][←][R] [ENTER] [▶] [→] =

Put together a vector with the two equations, using function →ARRY (find it using the command catalog [→] CAT) after typing a [2]:



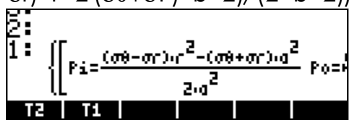


Now, suppose that we want to solve for P_i and P_o, given *a*, *b*, r, σ_{rr}, and $\sigma_{\theta\theta}$. We enter a vector with the unknowns:

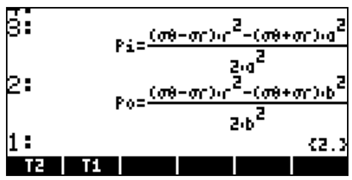


To solve for P_i and P_o, use the command SOLVE from the S.SLV menu (← S.SLV), it may take the calculator a minute to produce the result:

{['Pi=-(((σθ-σr)*r^2-(σθ+σr)*a^2)/(2*a^2))'
'Po=-(((σθ-σr)*r^2-(σθ+σr)*b^2)/(2*b^2))'] }, i.e.,



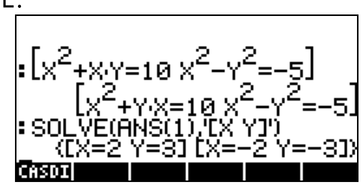
Notice that the result includes a vector [ ] contained within a list { }. To remove the list symbol, use EVAL. Finally, to decompose the vector, use function OBJ→. The result is:



These two examples constitute systems of linear equations that can be handled equally well with function LINSOLVE (see Chapter 11). The following example shows function SOLVE applied to a system of polynomial equations.

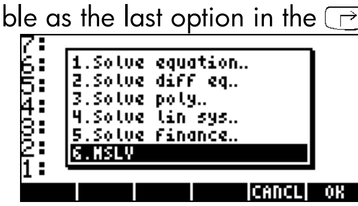
Example 3 - System of polynomial equations

The following screen shot shows the solution of the system $X^2+XY=10$, $X^2-Y^2=-5$, using function SOLVE:

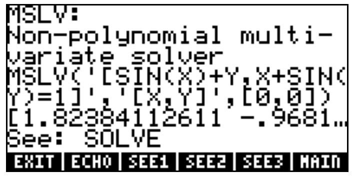


Solution to simultaneous equations with MSLV

Function MSLV is available as the last option in the → NUM.SLV menu:



The help-facility entry for function MSLV is shown next:

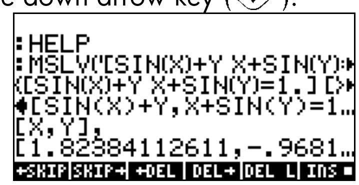


Example 1 – Example from the help facility

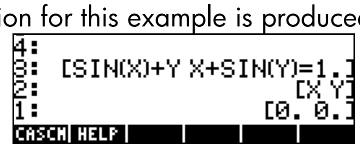
As with all function entries in the help facility, there is an example attached to the MSLV entry as shown above. Notice that function MSLV requires three arguments:

1. A vector containing the equations, i.e., '[SIN(X)+Y,X+SIN(Y)=1]'
2. A vector containing the variables to solve for, i.e., '[X,Y]'
3. A vector containing initial values for the solution, i.e., the initial values of both X and Y are zero for this example.

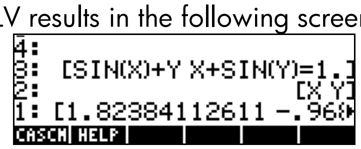
In ALG mode, press ECHO to copy the example to the stack, press ENTER to run the example. To see all the elements in the solution you need to activate the line editor by pressing the down arrow key (▼):



In RPN mode, the solution for this example is produced by using:



Activating function MSLV results in the following screen.



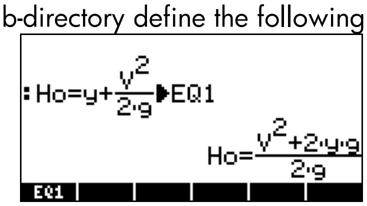
You may have noticed that, while producing the solution, the screen shows intermediate information on the upper left corner. Since the solution provided by MSLV is numerical, the information in the upper left corner shows the results of the iterative process used to obtain a solution. The final solution is *X = 1.8238, Y = -0.9681*.

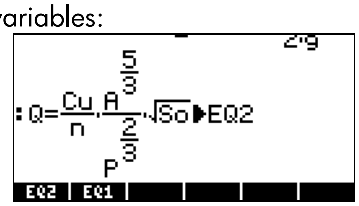
Example 2 - Entrance from a lake into an open channel

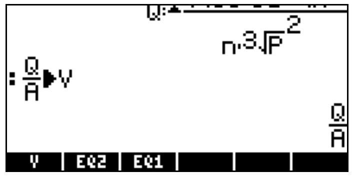
This particular problem in open channel flow requires the simultaneous solution of two equations, the equation of energy: $H_o = y + \frac{V^2}{2g}$, and Manning's equation: $Q = \frac{Cu}{n} \cdot \frac{A^{5/3}}{P^{2/3}} \cdot \sqrt{S_o}$. In these equations, H_o represents the energy head (m, or ft) available for a flow at the entrance to a channel, y is the flow depth (m or ft), V = Q/A is the flow velocity (m/s or ft/s), Q is the volumetric discharge (m^3/s or ft^3/s), A is the cross-sectional area (m^2 or ft^2), C_u is a coefficient that depends on the system of units (C_u = 1.0 for the SI, C_u = 1.486 for the English system of units), n is the Manning's coefficient, a measure of the channel surface roughness (e.g., for concrete, n = 0.012), P is the wetted perimeter of the cross section (m or ft), S_o is the slope of the channel bed expressed as a decimal fraction. For a trapezoidal channel, as shown below, the area is given by $A = (b + my)y$, while the wetted perimeter is given by $P = b + 2y\sqrt{1 + m^2}$, where b is the bottom width (m or ft), and m is the side slope (1V:mH) of the cross section.

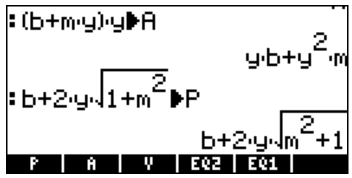
Typically, one has to solve the equations of energy and Manning's simultaneously for y and Q. Once these equations are written in terms of the primitive variables b, m, y, g, S_o, n, Cu, Q, and H_o, we are left with a system of equations of the form $f_1(y,Q) = 0$, $f_2(y,Q) = 0$. We can build these two equations as follows.

We assume that we will be using the ALG mode in the calculator, although defining the equations and solving them with MSLV is very similar in the RPN mode. Create a sub-directory, say CHANL (for open CHANneL), and within that sub-directory define the following variables:

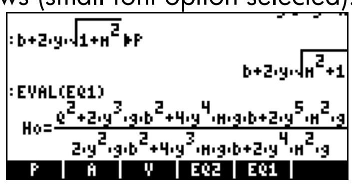
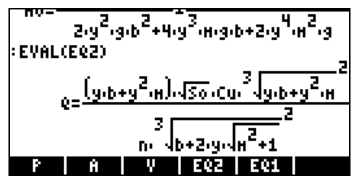






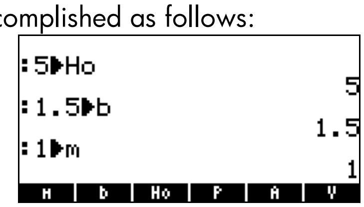


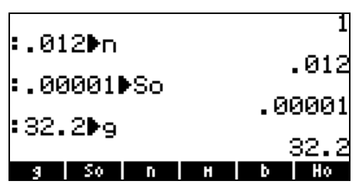
To see the original equations, EQ1 and EQ2, in terms of the primitive variables listed above, we can use function EVAL applied to each of the equations, i.e., EVAL EQ1 EVAL EQ2. The equations are listed in the stack as follows (small font option selected):

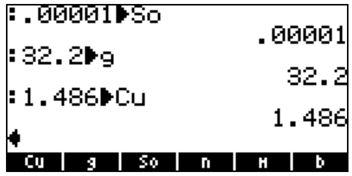



We can see that these equations are indeed given in terms of the primitive variables b, m, y, g, S_o, n, Cu, Q, and H_o.

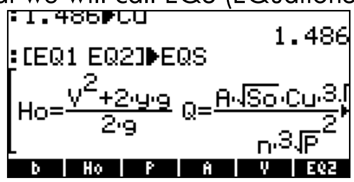
In order to solve for y and Q we need to give values to the other variables. Suppose we use H_0 = 5 ft, b = 1.5 ft, m = 1, n = 0.012, S_0 = 0.00001, g = 32.2, and Cu = 1.486. Before being able to use MSLV for the solution, we need to enter these values into the corresponding variable names. This can be accomplished as follows:



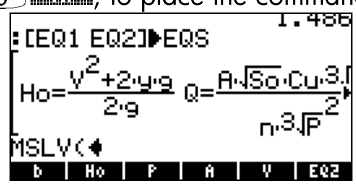




Now, we are ready to solve the equation. First, we need to put the two equations together into a vector. We can do this by actually storing the vector into a variable that we will call EQS (EQuationS):



As initial values for the variables y and Q we will use y = 5 (equal to the value of H_o, which is the maximum value that y can take) and Q = 10 (this is a guess). To obtain the solution we select function MSLV from the NUM.SLV menu, e.g., (→)NUM.SLV (6) [MSLV], to place the command in the screen:

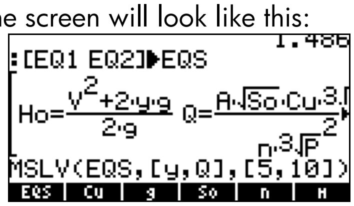


Next, we'll enter variable EQS: (NXT)(NXT) [EQS], followed by vector [y,Q]:

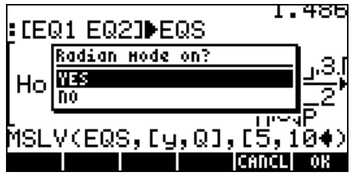
(→)(,)(←)([])(ALPHA)(←)(Y)(→)(,)(ALPHA)(Q)(▶)

and by the initial guesses (→)(,)(←)([]) (5)(→)(,) (1)(0).

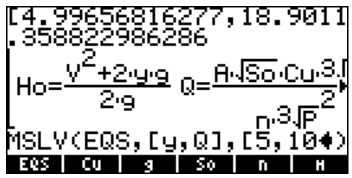
Before pressing (ENTER), the screen will look like this:



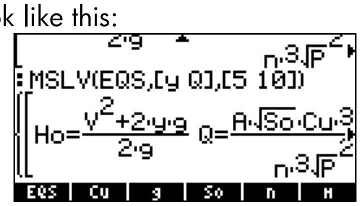
Press (ENTER) to solve the system of equations. You may, if your angular measure is not set to radians, get the following request:



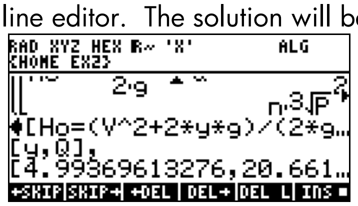
Press OK and allow the solution to proceed. An intermediate solution step may look like this:



The vector at the top representing the current value of [y,Q] as the solution progresses, and the value .358822986286 representing the criteria for convergence of the numerical method used in the solution. If the system is well posed, this value will diminish until reaching a value close to zero. At that point a numerical solution would have been found. The screen, after MSLV finds a solution will look like this:



The result is a list of three vectors. The first vector in the list will be the equations solved. The second vector is the list of unknowns. The third vector represents the solution. To be able to see these vectors, press the down-arrow key ▼ to activate the line editor. The solution will be shown as follows:



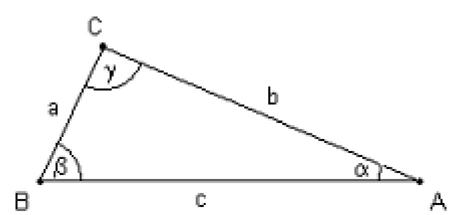
The solution suggested is [4.9936.., 20.661…]. This means, y = 4.99 ft, and Q = 20.661… ft^3/s. You can use the arrow keys (◀ ▶ ▲ ▼) to see the solution in detail.

Using the Multiple Equation Solver (MES)

The multiple equation solver is an environment where you can solve a system of multiple equations by solving for one unknown from one equation at a time. It is not really a solver to simultaneous solutions, rather, it is a one-by-one solver of a number of related equations. To illustrate the use of the MES for solving multiple equations we present an application related to trigonometry in the next section. The examples shown here are developed in the RPN mode.

Application 1 - Solution of triangles

In this section we use one important application of trigonometric functions: calculating the dimensions of a triangle. The solution is implemented in the calculator using the Multiple Equation Solver, or MES.
Consider the triangle ABC shown in the figure below.



The sum of the interior angles of any triangle is always 180°, i.e., $\alpha + \beta + \gamma = 180°$. The sine law indicates that:

$$\frac{\sin\alpha}{a} = \frac{\sin\beta}{b} = \frac{\sin\gamma}{c}.$$

The cosine law indicates that:

$$a^2 = b^2 + c^2 - 2 \cdot b \cdot c \cdot \cos\alpha,$$
$$b^2 = a^2 + c^2 - 2 \cdot a \cdot c \cdot \cos\beta,$$
$$c^2 = a^2 + b^2 - 2 \cdot a \cdot b \cdot \cos\gamma.$$

In order to solve any triangle, you need to know at least three of the following six variables: a, b, c, α, β, γ. Then, you can use the equations of the sine law, cosine law, and sum of interior angles of a triangle, to solve for the other three variables.

If the three sides are known, the area of the triangle can be calculated with Heron's formula $A = \sqrt{s \cdot (s-a) \cdot (s-b) \cdot (s-c)}$,where s is known as the semi-perimeter of the triangle, i.e., $s = \frac{a+b+c}{2}$.

Triangle solution using the Multiple Equation Solver (MES)
The Multiple Equation Solver (MES) is a feature that can be used to solve two or more coupled equations. It must be pointed out, however, that the MES does not solve the equations simultaneously. Rather, it takes the known variables, and then searches in a list of equations until it finds one that can be solved for one of the unknown variables. Then, it searches for another equation that can be solved for the next unknowns, and so on, until all unknowns have been solved for.

Creating a working directory
We will use the MES to solve for triangles by creating a list of equations corresponding to the sine and cosine laws, the law of the sum of interior angles, and Heron's formula for the area. First, create a sub-directory within HOME that we will call TRIANG, and move into that directory. See Chapter 2 for instructions on how to create a new sub-directory.

Entering the list of equations
Within TRIANG, enter the following list of equations either by typing them directly on the stack or by using the equation writer. (Recall that ALPHA ↱ A produces the character α, and ALPHA ↱ B produces the character β. The character γ needs to be ECHOed from ↱ CHARS):

'SIN(α)/a = SIN(β)/b'
'SIN(α)/a = SIN(γ)/c'
'SIN(β)/b = SIN(γ)/c'
'c^2 = a^2+b^2-2*a*b*COS(γ)'
'b^2 = a^2+c^2-2*a*c*COS(β)'
'a^2 = b^2+c^2-2*b*c*COS(α)'

$$\text{'}\alpha+\beta+\gamma = 180\text{'}$$
$$\text{'}s = (a+b+c)/2\text{'}$$
$$\text{'}A = \sqrt{(s*(s-a)*(s-b)*(s-c))}\text{'}$$

Then, enter the number [9], and create a list of equations by using: function →LIST (use the command catalog [→] _CAT). Store this list in the variable EQ.

The variable EQ contains the list of equations that will be scanned by the MES when trying to solve for the unknowns.

Entering a window title

Next, we will create a string variable to be called TITLE to contain the string "Triangle Solution", as follows:

[→] _""	Open double quotes in stack
[ALPHA] [ALPHA] [←] [ALPHA]	Locks keyboard into lower-case alpha.
[←] [T] [R] [I] [A] [N] [G] [L] [E] [SPC]	Enter text: Triangle_
[←] [S] [O] [L] [U] [T] [I] [O] [N]	Enter text: Solution
[ENTER]	Enter string "Triangle Solution" in stack
[']	Open single quotes in stack
[ALPHA] [ALPHA] [T] [I] [T] [L] [E] [ENTER]	Enter variable name 'TITLE'
[STO▸]	Store string into 'TITLE'

Creating a list of variables

Next, create a list of variable names in the stack that will look like this:

{ a b c α β γ A s }

and store it in variable LVARI (List of VARIables). The list of variables represents the order in which the variables will be listed when the MES gets started. It must include all the variables in the equations, or it will not work with function MITM (see below). Here is the sequence of keystrokes to use to prepare and store this list:

Press [VAR], if needed, to get your variables menu. Your menu should show the variables LVARI TITLE EQ .

Preparing to run the MES

The next step is to activate the MES and try one sample solution. Before we do that, however, we want to set the angular units to DEGrees, if they are not already set to that, by typing [ALPHA][ALPHA][D][E][G][ENTER].

Next, we want to keep in the stack the contents of TITLE and LVARI, by using:

[TITLE] [LVARI]

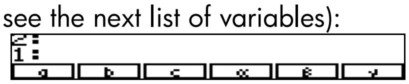
We will use the following MES functions

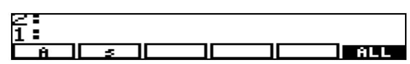
- MINIT: MES INITialization: initializes the variables in the equations stored in EQ.
- MITM: MES' Menu Item: Takes a title from stack level 2 and the list of variables from stack level 1 and places the title atop of the MES window, and the list of variables as soft menu keys in the order indicated by the list. In the present exercise, we already have a title ("Triangle Solution") and a list of variables ({ a b c α β γ A s }) in stack levels 2 and 1, respectively, ready to activate MITM.
- MSOLVR: MES SOLVER; activates the Multiple Equation Solver (MES) and waits for input by the user.

Running the MES interactively

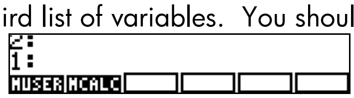
To get the MES started, with the variables TITLE and LVARI listed in the stack, activate command MINIT, then MITM, and finally, MSOLVR (find these functions in the catalog [→] _CAT).

The MES is launched with the following list of variables available (Press [NXT] to see the next list of variables):





Press [NXT] to see the third list of variables. You should see:



Press [NXT] once more to recover the first variable menu.

Let's try a simple solution of Case I, using a = 5, b = 3, c = 5. Use the following entries:

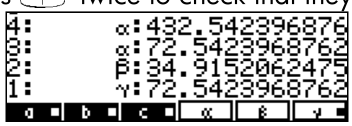
[5][a]   a:5 is listed in the top left corner of the display.

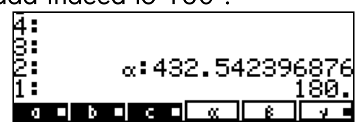
3 [b] b:3 is listed in the top left corner of the display.
5 [c] c:5 is listed in the top left corner of the display.
To solve for the angles use:
← [α] Calculator reports Solving for α, and shows the result α: 72.5423968763.

Note: If you get a value that is larger than 180, try the following:

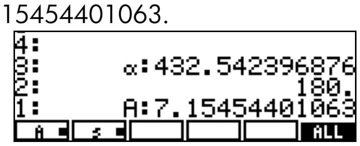
1 0 [α] Re-initialize a to a smaller value.
← [α] Calculator reports Solving for α

Next, we calculate the other two values:
← [β] The result is β: 34.9152062474 .
← [γ] The result is γ: 72.5423968763.
You should have the values of the three angles listed in stack levels 3 through 1. Press + twice to check that they add indeed to 180°.



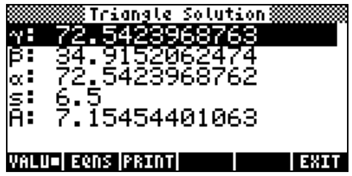


Press NXT to move to the next variables menu. To calculate the area use: ← [A]. The calculator first solves for all the other variables, and then finds the area as A: 7.15454401063.



Note: When a solution is found, the calculator reports the conditions for the solution as either Zero, or Sign Reversal. Other messages may occur if the calculator has difficulties finding a solution.

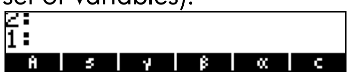
Pressing ← ALL will solve for all the variables, temporarily showing the intermediate results. Press → ALL to see the solutions:



When done, press [ON] to return to the MES environment. Press [VAR] to exit the MES environment and return to the normal calculator display.

Organizing the variables in the sub directory

Your variable menu will now contain the variables (press [NXT] to see the second set of variables):



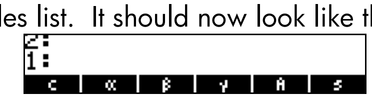
Variables corresponding to all the variables in the equations in EQ have been created. There is also a new variable called *Mpar* (MES parameters), which contains information regarding the setting up of the MES for this particular set of equations. If you use [→] [Mpar] to see the contents of the variable *Mpar*. You will get the cryptic message: Library Data. The meaning of this is that the MES parameters are coded in a binary file, which cannot be accessed by the editor.

Next, we want to place them in the menu labels in a different order than the one listed above, by following these steps:

1. Create a list containing { EQ Mpar LVARI TITLE }, by using:
   [←] { } [EQ] [Mpar] [LVARI] [TITLE] [ENTER]
2. Place contents of LVARI in the stack, by using: [LVARI].
3. Join the two lists by pressing [+].

Use function ORDER (use the command catalog [→] CAT) to order the variables as shown in the list in stack level 1.

4. Press [VAR] to recover your variables list. It should now look like this:



5. Press [NXT] to recover the first variable menu.

Programming the MES triangle solution using User RPL

To facilitate activating the MES for future solutions, we will create a program that will load the MES with a single keystroke. The program should look like this: << DEG MINIT TITLE LVARI MITM MSOLVR >>, and can be typed in by using:

[→] « »	Opens the program symbol
[ALPHA] [ALPHA]	Locks alphanumeric keyboard
[D] [E] [G] [SPC]	Type in DEG (angular units set to DEGrees)
[M] [I] [N] [I] [T] [SPC]	Type in MINIT_
[ALPHA]	Unlocks alphanumeric keyboard
[TITLE]	List the name TITLE in the program
[LVARI]	List the name LVARI in the program
[ALPHA] [ALPHA]	Locks alphanumeric keyboard
[M] [I] [T] [M] [SPC]	Type in MITM_
[M] [S] [O] [L] [V] [R]	Type in MSOLVR
[ENTER]	Enter program in stack

Store the program in a variable called TRISOL, for TRIangle SOLution, by using: ['] [ALPHA] [ALPHA] [T] [R] [I] [S] [O] [L] [ENTER] [STO▶]

Press [VAR], if needed, to recover your list of variables. A soft key label [TRISO] should be available in your menu.

Running the program – solution examples

To run the program, press the [TRISO] soft menu key. You will now have the MES menu corresponding to the triangle solution. Let's try examples of the three cases listed earlier for triangle solution.

Example 1 – Right triangle

Use a = 3, b = 4, c = 5. Here is the solution sequence:

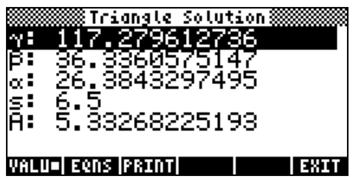
[3] [a] [4] [b] [5] [c]	To enter data
[←] [α]	The result is α: 36.8698976458
[←] [β]	The result is β: 53.1301023541.
[←] [γ]	The result is γ: 90.
[NXT]	To move to the next variables menu.
[←] [A]	The result is A: 6.
[NXT] [NXT]	To move to the next variables menu.

Example 2 - Any type of triangle

Use a = 3, b = 4, c = 6. The solution procedure used here consists of solving for all variables at once, and then recalling the solutions to the stack:

VAR TRISO	To clear up data and re-start MES
3 [a] 4 [b] 6 [c]	To enter data
NXT	To move to the next variables menu.
← ALL	Solve for all the unknowns.
→ ALL	Show the solution:

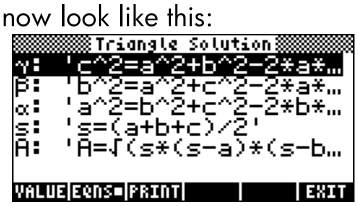
The solution is:



At the bottom of the screen, you will have the soft menu keys:

VALU▪ EQNS PRINT EXIT

The square dot in VALU▪ indicates that the values of the variables, rather than the equations from which they were solved, are shown in the display. To see the equations used in the solution of each variable, press the EQNS soft menu key. The display will now look like this:



The soft menu key PRINT is used to print the screen in a printer, if available. And EXIT returns you to the MES environment for a new solution, if needed. To return to normal calculator display, press VAR.

The following table of triangle solutions shows the data input in bold face and the solution in italics. Try running the program with these inputs to verify the solutions. Please remember to press VAR TRISO at the end of each solution to clear up variables and start the MES solution again. Otherwise, you may carry over information from the previous solution that may wreck havoc with your current calculations.

a	***b***	***c***	$\alpha(^o)$	$\beta(^o)$	$\gamma(^o)$	***A***
2.5	*6.9837*	**7.2**	*20.299*	**75**	*84.771*	*8.6933*
7.2	**8.5**	*14.26*	*22.616*	**27**	*130.38*	*23.309*
21.92	**17.5**	**13.2**	**90**	*52.97*	**37.03**	*115.5*
41.92	**23**	*29.6*	**75**	**32**	*73*	*328.81*
10.27	*3.26*	**10.5**	*77*	**18**	**85**	*16.66*
17	**25**	**32**	*31.79*	*50.78*	*97.44*	*210.71*

Adding an INFO button to your directory

An information button can be useful for your directory to help you remember the operation of the functions in the directory. In this directory, all we need to remember is to press TRISO to get a triangle solution started. You may want to type in the following program: <<"Press [TRISO] to start." MSGBOX >>, and store it in a variable called INFO. As a result, the first variable in your directory will be the INFO button.

Application 2 - Velocity and acceleration in polar coordinates

Two-dimensional particle motion in polar coordinates often involves determining the radial and transverse components of the velocity and acceleration of the particle given r, r' = dr/dt, r" = d^2r/dt^2, θ, θ' = d θ /dt, and, θ" = $d^2\theta/dt^2$. The following equations are used:

$$v_r = \dot{r} \qquad a_r = \ddot{r} - r\dot{\theta}^2$$

$$v_\theta = r\dot{\theta} \qquad a_\theta = r\ddot{\theta} + 2\dot{r}\dot{\theta}$$

Create a subdirectory called POLC (POLar Coordinates), which we will use to calculate velocities and accelerations in polar coordinates. Within that subdirectory, enter the following variables:

Program or value	Store into variable:
<< PEQ STEQ MINIT NAME LIST MITM MSOLVR >>	**SOLVEP**
"vel. & acc. polar coord."	**NAME**
{ r rD rDD θD θDD vr vθ v ar aθ a }	**LIST**
{ 'vr = rD' 'vθ = r*θD' 'v = √(vr^2 + vθ^2)' 'ar = rDD – r*θD^2' 'aθ = r*θDD + 2*rD*θD' 'a = √(ar^2 + aθ^2)' }	**PEQ**

An explanation of the variables follows:

SOLVEP = a program that triggers the multiple equation solver for the particular set of equations stored in variable **PEQ**;

NAME = a variable storing the name of the multiple equation solver, namely, *"vel. & acc. polar coord."*;

LIST = a list of the variable used in the calculations, placed in the order we want them to show up in the multiple equation solver environment;

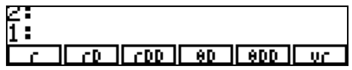
PEQ = list of equations to be solved, corresponding to the radial and transverse components of velocity (**vr**, **v**θ) and acceleration (**ar**, **a**•) in polar coordinates, as well as equations to calculate the magnitude of the velocity (**v**) and the acceleration (**a**) when the polar components are known.

r, **rD**, **rDD** = r (radial coordinate), r-dot (first derivative of r), r-double dot (second derivative of r).

θ**D**, θ**DD** = θ-dot (first derivative of θ), θ-double dot (second derivative of θ).

Suppose you are given the following information: r = 2.5, rD = 0.5, rDD = -1.5, θD = 2.3, θDD = -6.5, and you are asked to find vr, vθ, ar, aθ, v, and a.

Start the multiple equation solver by pressing VAR [SOLVE]. The calculator produces a screen labeled , *"vel. & acc. polar coord."*, that looks as follows:

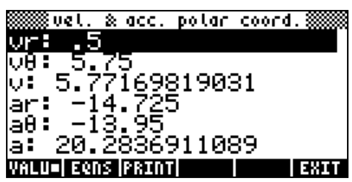


To enter the values of the known variables, just type the value and press the button corresponding to the variable to be entered. Use the following keystrokes: 2.5 [r] 0.5 [rD] 1.5 +/- [rDD] 2.3 [θD] 6.5 +/- [θDD].

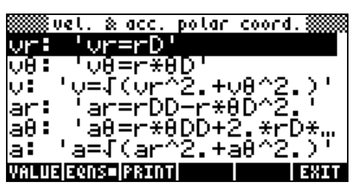
Notice that after you enter a particular value, the calculator displays the variable and its value in the upper left corner of the display. We have now entered the known variables. To calculate the unknowns we can proceed in two ways:

a). Solve for individual variables, for example, ← [vr] gives vr: 0.500. Press NXT ← [vθ] to get vθ : 5.750 , and so on. The remaining results are v: 5.77169819031; *a*r: -14.725; *a*θ: -13.95; and *a*: 20.2836911089.; or,

b). Solve for all variables at once, by pressing ← [ALL]. The calculator will flash the solutions as it finds them. When the calculator stops, you can press → [ALL] to list all results. For this case we have:



Pressing the soft-menu key [EQNS] will let you know the equations used to solve for each of the values in the screen:



To use a new set of values press, either EXIT ALL NXT NXT, or VAR SOLVE.

Let's try another example using r = 2.5, vr = rD = -0.5, rDD = 1.5, v = 3.0, a = 25.0. Find, θD, θDD, vθ, ar, and aθ. You should get the following results: