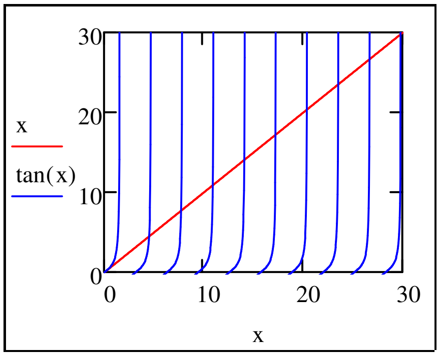
SOLVING NONLINEAR EQUATIONS

Basic use of the "root" function

A quite common problem in mathematics is to find the values of x where tan(x) = x. This is a nonlinear equation, since tan is a nonlinear function. The problem cannot be solved analytically, but it can be solved numerically using Mathcad.

First you want to graph the functions x and tan(x):



You see there is one simple solution, or "root", namely x=0. tan(0) = 0. You also see that the other roots are pretty close to 3π/2, 5π/2, 7π/2, ...

To have mathcad find the roots, first define a function that is *zero* at the roots. The simplest way to do that is to take everything to the same side in the equation tan(x) = x:

$\boxed{f_{simple}(x) := \tan(x) - x}$ Put a block around it.

Now let's try to get the root close to 3π/2. To do so, we must give Mathcad an "initial guess". We must give it a starting value of x, from which the start searching for the root. Normally the closer the initial guess is to the actual root, the better. But while 3π/2 would be very close to the root, it is not a good idea to give that. The tan(x) is singular, infinite, at 3π/2. Even if the search algorithm does not crash evaluating f_1 at 3π/2, it would be just as likely to find the root at 5π/2 instead of the one at 3π/2. Try starting at π instead:

$$xx := \pi$$

Now use the "root" function to find the root:

$$x_1 := \mathrm{root}\left(f_{simple}(xx), xx\right) \qquad x_1 = 3.135 \times 10^{-3}$$

Oops. Mathcad found the wrong root! It found the root at 0. And not to great accuracy too! Mathcad has a tolerance for errors while solving equations, stored in a variable called TOL, and it is pretty liberal:

$$TOL = 1 \times 10^{-3}$$

Making this a lot smaller should improve the root at 0:

$$TOL := 10^{-10} \qquad TOL = 1 \times 10^{-10}$$

$$xx := \pi \qquad x_0 := \mathrm{root}\left(f_{simple}(xx), xx\right) \qquad x_0 = 7.807 \times 10^{-9} \qquad f_{simple}\left(x_0\right) = 0$$

Note that while the root is not exactly zero, f_{simple} is. For small x, f_{simple} is approximately $x^3/3$, and that is apparently small enough that Mathcad truncates it to 0.

But we wanted the root near 3π/2. Let's try a new starting point:

$xx := 3\frac{\pi}{2} - 0.0000001 \quad x_1 := \mathrm{root}(f_{simple}(xx), xx) \quad x_1 = -1.056 \times 10^{-4}$ curse

$xx := 3\frac{\pi}{2} - 0.0001 \quad x_1 := \mathrm{root}(f_{simple}(xx), xx) \quad x_1 = -1.894 \times 10^{-5}$ curse

$xx := 3\frac{\pi}{2} - 0.01 \quad x_1 := \mathrm{root}(f_{simple}(xx), xx) \quad x_1 = 4.493$ got it!

This is a mess. We can make it simpler by being a bit clever: tan(x) = sin(x) /cos(x), so we can rewrite tan(x)=x to sin(x)=x cos(x) and then bring everything to the same side. Bingo! No more singularities, so we should be able to start from 3π/2 to find root x_1:

$$f_{clever}(x) := \sin(x) - x\cdot\cos(x)$$

Try that one:

$xx := 3\frac{\pi}{2} \quad x_1 := \mathrm{root}(f_{clever}(xx), xx) \quad x_1 = 4.493 \quad f_{clever}(x_1) = 0$

$xx := 5\frac{\pi}{2} \quad x_2 := \mathrm{root}(f_{clever}(xx), xx) \quad x_2 = 7.725 \quad f_{clever}(x_2) = -2.998 \times 10^{-15}$

$xx := 7\frac{\pi}{2} \quad x_3 := \mathrm{root}(f_{clever}(xx), xx) \quad x_3 = 10.904 \quad f_{clever}(x_3) = 9.104 \times 10^{-15}$

That was much easier! We can also readily find the root at zero:

$xx := 0.5 \quad x_0 := \mathrm{root}(f_{clever}(xx), xx) \quad x_0 = 8.18 \times 10^{-7} \quad f_{clever}(x_0) = 0$

The Mathcad "root" function will give you the roots of an equation if you give it a starting value close enough to the root, and beware of singularities.

Using "root" with an interval.

We can fix the problems with f_{simple} in another way. If you look back at the graph, you will see that f_{simple} is negative for x = π, but positive for x very slightly less than 3π/2. And there is just one root in between. In that case you can *force* root to search only in that interval by specifing its end points:

(no initial guess needed) $x_1 := \mathrm{root}\left(f_{simple}(x), x, \pi, 3\frac{\pi}{2} - 0.00000001\right)$ $\qquad x_1 = 4.493$

This is absolutely reliable. For example, try 0:

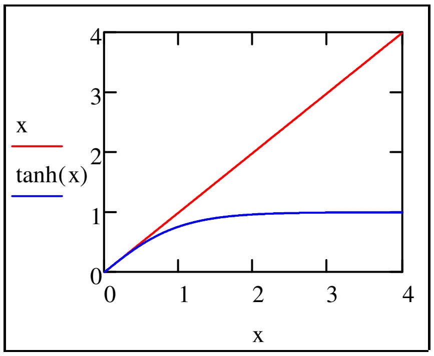
$x_0 := \mathrm{root}\left(f_{simple}(x), x, -.5, .5\right)$ $\qquad x_0 = 0$

Using root with and interval will always work provided that:
1. The end point values of the function have different sign.
2. There is only one change of sign of the function in the interval. (If there is more than one change of sign, root will find one of them but which one is uncertain.)
3. There is only one real unknown. Intervals do not work for complex or multiple unknowns. And root does not work with multiple unknowns anyway.

Using "root" with an interval, if possible, will get rid of headaches.

Complex roots using "root".

Consider the equation tanh(x) = x. This has only one real root, x = 0.



But since tanh(iy)=i tan(y), there are infinitely many roots of the form i 3π/2, i 5π/2, ... and their complex conjugates.

$$\mathrm{fth}(x) := \sinh(x) - x\cdot\cosh(x)$$

Trying starting points like 1, 2, -1, -2, 5, -5, ...,you always get the trivial root 0:

$xx := 5 \qquad xth := \mathrm{root}(\mathrm{fth}(xx), xx) \qquad xth = 2.204\times 10^{-5}$

So try a complex starting point

$xx := 3\cdot\frac{\pi}{2}\cdot i \qquad xth := \mathrm{root}(\mathrm{fth}(xx), xx) \qquad xth = 4.493i$

Note: The secant method used by the root function would not start looking for complex roots given a real equation and real starting point. Assuming that the Mueller method also used by root according to the Mathcad help is the Muller-Traub method, that one might. Giving a complex starting position like above might be more reliable, though.

Provide complex initial guesses for complex roots.

Roots of polynomials using "polyroots"

For polynomial equations, you want to use a dedicated polynomial root finder. Mathcad provides polyroot. As an example consider the quartic polynomial:

$$p(x) := x^4 - 2\cdot x^3 + 22\cdot x - 9$$

We want to know the roots of this polynomial. First put the coefficients into a vector, starting with the coefficient of x^0:

$$\text{coefs} := \begin{pmatrix} -9 \\ 22 \\ 0 \\ -2 \\ 1 \end{pmatrix}$$

Now use function polyroots

$$\text{roots} := \text{polyroots}(\text{coefs}) \qquad \text{roots} = \begin{pmatrix} -2.414 \\ 0.414 \\ 2 - 2.236i \\ 2 + 2.236i \end{pmatrix}$$

Those are the correct 4 roots: p(x) was chosen to be $\left[(x+1)^2 - 2\right]\left[(x-2)^2 + 5\right]$, and that has roots -1 - $\sqrt{2}$, -1 + $\sqrt{2}$, 2 - i$\sqrt{5}$, and 2+ i$\sqrt{5}$.

Use "polyroots" to find roots of polynomials.

Solution of nonlinear systems using a "Given / ACTION" block

A Given / ACTION block allows much more complicated nonlinear systems to be solved, possibly under additional inequality constraints. Or the minima or maxima of variables described by such systems to be found. Or you can even solve ordinary and partial differential equations with such blocks.

Consider first how a previous problem solved with "root" can also be solved with such a Given / Find block:

$xx := -3\frac{\pi}{2}$ You must again provide an initial guess to the solution.

Given This must be a *math* word. *Do not press space.*

$\sin(xx) - xx\cdot\cos(xx) = 0$ This is *not evaluation*: use Ctrl+=, not =.

$x_1 := \text{Find}(xx)$ This is the ACTION that terminates the block.

$\boxed{x_1 = -4.493}$ Print it out. Yes, that is the right solution!

Next let's try a system. Like finding the intersection points of an ellipse and a circle:

$xx := 2$ $yy := -2$ Initial guesses for the x and y positions.

Given

$xx^2 + xx\cdot yy + yy^2 = 4$ The equation of the ellipse.

$(xx-2)^2 + (yy+2)^2 = 1$ The equation of the circle.

$sol := \text{Find}(xx, yy)$ This is the ACTION that terminates the block.

$sol = \begin{pmatrix} 2.306 \\ -1.048 \end{pmatrix}$ $\boxed{ORIGIN := 1}$ $\boxed{xx := sol_1 = 2.306}$ $\boxed{yy := sol_2 = -1.048}$

Sounds about right.

Consider next a problem of my graduate math class; finding the shortest ladder to reach a house over an 8 ft wall in front of it. Note: do not use units.

$d := \frac{27}{8}$ — Initial guess for the distance from the house where the ladder is placed on the ground.

$h := 8$ — Initial guess of the height where the ladder hits the house.

$l(h, d) := \sqrt{h^2 + d^2}$ — Length of the ladder. *This must be outside the block*.

Given

$h \cdot \left(d - \frac{27}{8}\right) - 8d \geq 0$ — The constraint that the ladder must go over the 8 ft wall, expressed as an inequality.

$h > 8$

$d > \frac{27}{8}$

These two inequalities are needed to prevent Mathcad coming up with the trivial solution h = d = 0.

$sol := \text{Minimize}(l, h, d)$ — This is the ACTION that terminates the block.

$sol = \begin{pmatrix} 12.5 \\ 9.375 \end{pmatrix}$ $\quad ORIGIN := 1 \quad \boxed{h := sol_1 = 12.5} \quad \boxed{d := sol_2 = 9.375}$

Exact: h=12.5 Exact: d=9.375