

**CLIVE MOLER:** The Lotka-Volterra predator-prey equations are the granddaddy of all models involving competition between species. They are the foundation of fields like mathematical ecology. Think of the two species as rabbits and foxes or moose and wolves or little fish in big fish.

$y_1$  represents the prey, who would live peacefully by themselves if there were no predators. I've chosen the units of time and population so that the coefficients in front of the leading linear terms are one. So  $y_1'$  equals  $y_1$  represents exponential growth of the prey in the absence of any predators.

The predators need the prey, live on the prey. So in the absence of any prey, this minus sign is all important. So  $y_2'$  equals  $-y_2$  represents exponential decay. And the predators die off exponentially in the absence of any prey.

But then here are the non-linear terms. These are like logistics terms, except with the interaction between the two species. The growth of  $y_1$  is limited by the presence of  $y_2$ . So  $y_1$  will grow until this term becomes 0 one  $y_2$  reaches  $\mu_2$ .

On the other hand, the decay of  $y_2$  becomes 0 when  $y_1$  reaches  $\mu_1$ . To complete this specification, we need the initial conditions. So we have two values  $\eta_1$  and  $\eta_2$ , which are the initial values of  $y_1$  and  $y_2$ .

So these four parameters, two  $\mu$ s and two  $\eta$ s, are the four parameters we have in our predator-prey model. Don't worry about the fact that these are continuous variables and that we can have non-integer numbers of individuals.

We can have half of a rabbit or a tenth of a moose. These are, after all, models that are idealized versions of what's happening in nature. The critical points are when the derivatives become 0. There's a critical point at the origin.

But the interesting one is when these terms become 0. So that's the point where  $y_1$  and  $y_2$  are equal to the  $\mu_1$  and  $\mu_2$ . We have to look at the Jacobian. And here's the Jacobian in general. And at the critical point, the Jacobian-- here's the Jacobian.

The eigenvalues of the Jacobian are plus or minus  $i$ . And so this critical point is a stable center with a period  $2\pi$ . So these are non-linear equations. We can't express the solution in terms of

simple analytic functions.

We have to compute them numerically. But we do know this about their behavior. If the initial conditions are close to the critical point, the solution is periodic. The period is close to  $2\pi$ . And the orbit is close to an ellipse.

On the other hand, if the initial conditions are far from the critical point, then it turns out the solution is still periodic. But the period is greater than  $2\pi$  and the orbit is far from an ellipse. OK, let's bring up MATLAB and compute a solution.

We need the parameters. Here's a  $\mu$  and an  $\eta$ . And I'll set the differential equation. And then choose ODE 45 and we'll integrate from 0 to 25. And here's the solution. It's periodic. A predator and a prey. And it looks like the period, it's returning back to the initial condition is 100 and 400.

And it's returning back to that along about here. And we've integrated over something more than three periods. I happen to know that the period is 6.5. And so I can-- I'm going to set-- I want to compute to higher accuracy.  $10^{-6}$ .

And let's capture the solution and integrate over three periods. It generated 337 points. Let's plot it with the finer dots. And we can see here, we've gone over three periods and return back to our initial values of 100 and 300.

And now I'm going to use something that'll show off the periodicity of function in MATLAB called Comet. One orbit. Two orbits. Three orbits. Determining the period of a periodic solution is often the important part of a calculation.

In the MATLAB ODE suite, this is done with an event handler. So I'm going to use ODE's set to provide an event handler called Pit Stop. And here's the code in Pit Stop. This code is called every time step in the integration.

And I'm not going to go into detail here, but it computes a function  $g$  that we're looking to see when this is 0. And when  $g$  returns to 0, the integration is stopped. I'm going to-- I have a display function in here.

Ordinarily, this wouldn't be here. But I want to look at this to see how the calculation is done. It says, terminate when  $y$  returns to the point where its angle  $\mu$  is the same as the angle between  $\eta$  and  $\mu$ . This is more reliable than just finding when the solution returns back to

its initial condition.

So let's run ODE 45 and tell it to integrate over an infinite time step, over an infinite interval. That's not going to happen, because we're going to stop as soon as  $g$  is 0 but with the option vector set with this event handler. So here it is.

And here's the output produced by Pit Stop as we integrate along. Here's the values that it's found. And here's where  $g$  returns to 0. I've produced a  $t$  vector with 117 values in it. And the final value of  $t$  is this value that I said was the period of this solution.

So that's how the period was determined by this event handling feature in the ODE solvers. Great. I have a program called Predator Prey that's in the collection of programs that comes with NCM, Numerical Computing with MATLAB. My book that's available on the MathWorks website.

Predator prey offers this graphic user interface to demonstrate what we've been talking about the predator prey equations. The top display shows the phase plane plot, the plot of prey versus predator. And the bottom display shows the time series plot, the plot of the two populations.

And initially, it's set at the conditions we've been talking about. Here's 400 rabbits and 100 foxes around the critical point of 300 rabbits and 200 foxes. And here's the period of 6.5 some odd that we've been working with.

Now, it says drag either dot. And you can change the initial conditions or the critical point. If I bring the initial conditions close to the critical point, then the phase plane plot becomes close to ellipse. And the period becomes close to  $2\pi$ .

This is 6.28, which is twice 3.14. But if I change it so that the two are far away from each other, then the phase plane plot becomes quite different from an ellipse. It's always periodic. That's an amazing thing about these nonlinear equations.

They always have a periodic solution. But now as you can see, the period is greater than  $2\pi$ . And the solutions are very far from sinusoids. So that's the pred prey app that's available with the programs in from the mathworks website under NCM for Numerical Computing with MATLAB.

Whoops, I stand corrected. If you Google Moler predprey, it tries to talk me out of it, but then it

shows it's in my second book, Experiments with MATLAB. There are two books. Numerical Computing with MATLAB and Experiments with MATLAB.

And pred prey is in the second one, Experiments with MATLAB. You can go to website and you can download all of the programs from EXM or you can go down here and here's pred prey. So it's in Experiments with MATLAB on the MathWorks website. Just Google Moler predator prey and you'll find it.