RY is the relative yield, which is the rate of pasture growth expressed as a fraction of the rate of growth possible when soil phosphorus and sulphur status are both high. The expression for RY is the multiple of two Mitscherlich curves, one limited by P availability, and one limited by S availability:

```
> RYp:=1-exp(-kP*soilP):
> RYs:=1-exp(-kS*soilS):

> RY=RYp*RYs;
> # we do not assign RY at this stage
```

$$RY = \left(1 - e^{(-kP\ soilP)}\right) \left(1 - e^{(-kS\ soilS)}\right)$$

Fertilizer is assumed to increase fertility immediately after it is applied. Therefore, the effective levels of P and S which affect yield are:

```
> soilP:=P+FP;
> soilS:=S+FS+rho*(E0+FE0)+(kQ/kS)*Q;
```

$$\begin{aligned} soilP &:= P + FP \\ soilS &:= S + FS + \rho\,(E0 + FE0) + \frac{kQ\ Q}{kS} \end{aligned}$$

Even though plants can take up only inorganic sulphur, the observed relationship between pasture growth and soil sulphur is more reliable when related to organic sulphur. This is why Q has been used to drive the sulphur response. Organic sulphur is converted to its equivalent sulphate form by multiplication by a factor of (kQ/kS).

Soil sulphur fertility is described by three variables, soil organic sulphur (Q), extra soil sulphate (S), and unoxidized elemental sulphur ($E0$). Each of these variables has an equation describing the cycling. The simplest is that for elemental sulphur, $E0$. Additions to elemental sulphur come from elemental sulphur fertilizer applications, $FE0$. Each year a proportion (ρ) of this oxidizes to become sulphate sulphur. So the accumulation of $E0$ from year to year is

```
> EE0:=-rho*(E0+FE0)+FE0;
```

$$EE0 := -\rho\,(E0 + FE0) + FE0$$

Sulphate sulphur dynamics are similar, with contributions from fertilizer (FS), oxidation of elemental sulphur, and rain, Z. The main process affecting the loss of sulphate is leaching, λS, which is driven by rainfall and soil type.

```
> ES:=-lambda*(S+FS+rho*(E0+FE0))+Z+FS+rho
> *(E0+FE0);
```

$$ES := -\lambda\,(S + FS + \rho\,(E0 + FE0)) + Z + FS + \rho\,(E0 + FE0)$$

Finally, organic sulphur dynamics consist of only two terms: losses due to mineralisation, and return of organic sulphur in dung and leaf litter, which is proportional to the yield:

```
> EQ:=-mu*Q+g*RY;
```

$$EQ := -\mu\,Q + g\,RY$$

We have now described the carryover of the four nutrients: P, $E0$, S, and Q. The software selects values for the parameters from look-up tables keyed to fertilizer type (ρ), pasture type (kP, hP, kS, kQ), soil type (β, Ps, μ, λ), stock class (g, $a1$, $a2$), land slope (λ) and climatic zone (Z, λ). These equations give sufficient information to simulate the dynamics of the soil nutrients from year to year into the future.

Economics

Once the increases and decreases in nutrient availability and pasture yield are predicted, it is necessary to calculate the economic costs and benefits. The aim is to calculate the optimal application of fertilizer (FP, $FE0$, FS) to maximize long term profit from the farm. The revenue from the animals, which is assumed to be proportional to the amount of pasture grown, and the cost of applying the fertilizer, are respectively:

```
> revenue:=r*RY;
> cost:=cP*FP+cS*FS+cE0*FE0;
```

$$\begin{aligned} revenue &:= r\ RY \\ cost &:= cP\ FP + cS\ FS + cE0\ FE0 \end{aligned}$$

We assume that the costs are incurred at the beginning of each year, and that the revenue is received at the close of that year. An economic discount factor, α, is defined as $1/(1+i)$ where i is the interest rate from alternative investments. Maximising profit from the farm then consists of maximising the "net present value" (NPV) of costs and revenue discounted and summed over future years [2]:

```
> NPVt:=alpha^t*revenue-alpha^(t-1)*cost:
> NPV:=Sum(NPVt,t=1..T);
```

$$NPV := \sum_{t=1}^{T} \left(\alpha^t\ r\ RY - \alpha^{(t-1)}\,(cP\ FP + cS\ FS + cE0\ FE0)\right)$$

Optimal control solution

The optimal control problem consists of choosing the fertilizer applications in each year to maximize NPV, such that the dynamics of the state variables, P, $E0$, S, and Q are governed by the equations EP, $EE0$, ES, and EQ described above. In addition a budgetary constraint is introduced, so

that the total fertilizer expenditure in any one year may not exceed d: This constraint enables a simple form of risk management to be practised.

```
> constraint:=d-cost;
```

$$constraint := d - cP\,FP - cS\,FS - cE0\,FE0$$

The multi-variable optimal control method for discrete-time models is given in Clark [1], and consists of forming the Hamiltonian function, H, and then solving the "Maximum Principle" and the "Adjoint Equations", which are based on partial derivatives of H. The Hamiltonian function is constructed by appending the state equations (eg. EP) and the constraint to the objective function (NPV). The adjoint multipliers adP, $adE0$, adS, and adQ are introduced, as well as the Lagrange multiplier, ε.

```
> H:=NPV+adP*EP+adE0*EE0+adS*ES+adQ*EQ+
> epsilon*constraint;
```

$$\begin{aligned} H := & \left(\sum_{t=1}^{T} \left(\alpha^{t}\, r\, RY \right.\right. \\ & \left.\left. -\alpha^{(t-1)} \left(cP\,FP + cS\,FS + cE0\,FE0 \right) \right) \right) \\ & + adP \left(-\beta \left(P + FP \right) + Ps + FP \right. \\ & -\left(a1 \left(1 - e^{\left(-hP \left(P+FP \right) \right)} \right) + a2 \right) RY \Big) \\ & + adE0 \left(-\rho \left(E0 + FE0 \right) + FE0 \right) \\ & + adS \left(-\lambda \left(S + FS + \rho \left(E0 + FE0 \right) \right) \right) \\ & + Z + FS + \rho \left(E0 + FE0 \right) \Big) \\ & + adQ \left(-\mu\, Q + g\, RY \right) \\ & + \varepsilon \left(d - cP\,FP - cS\,FS - cE0\,FE0 \right) \end{aligned}$$

The problem is quite complicated, and in the interests of finding an approximate solution, it will be assumed that both elemental sulphur ($E0$) and sulphate (S) are in equilibrium at all points in time, ie.,

```
> E0:=solve(EE0=0,E0);
> S:=solve(ES=0,S);
```

$$E0 := \frac{-\rho\,FE0 + FE0}{\rho}$$

$$S := \frac{-\lambda\,FS - \lambda\,FE0 + Z + FS + FE0}{\lambda}$$

Also, because we intend to look for equilibrium solutions, where the fertilizer rates and nutrient levels are the same from year to year, the dependence of the Hamiltonian on time (t) is dropped. Now defining RY at last, the simplified Hamiltonian is:

```
> t:=1:
> H:=NPVt+adP*EP+adQ*EQ+epsilon*constraint;
> H:=subs(RY=RYp*RYs,H):
```

$$H := \alpha\, r\, RY - cP\,FP - cS\,FS - cE0\,FE0$$

$$\begin{aligned} & + adP \left(-\beta \left(P + FP \right) + Ps + FP \right. \\ & -\left(a1 \left(1 - e^{\left(-hP \left(P+FP \right) \right)} \right) + a2 \right) RY \Big) \\ & + adQ \left(-\mu\, Q + g\, RY \right) \\ & + \varepsilon \left(d - cP\,FP - cS\,FS - cE0\,FE0 \right) \end{aligned}$$

The first part of the optimal control solution is to solve the Maximum Principle. That is, the control variables FP and FS must maximize H. Since the constraints operate on FP and FS, this is straightforward — if the constraints are not tight, H is maximized when its partial derivatives with respect to FP and FS are zero:

```
> maxP:={diff(H,FP)=0,diff(H,FS)=0}:
```

(From this point on, the results are generally too large to display). The Maximum Principle equations are linear in the adjoint variables adP and adQ, and we solve for these.

```
> advars:={adP,adQ};
> advarsx:=solve(maxP,advars):
```

$$advars := \{ adP, adQ \}$$

The second step is to solve the adjoint equations, where the partial derivatives of H with respect to the adjoint variables are equated to the change in the adjoint variables from year to year [1, p.236]. Since we are looking for equilibrium solutions, the adjoint variables decline by a factor of α each year [1, p.237]. We substitute the expressions for the adjoint variables already obtained:

```
> adeq1:=subs(advarsx,-diff(H,P)=adP
> *(1-alpha^(-1))):
> adeq2:=subs(advarsx,-diff(H,Q)=adQ
> *(1-alpha^(-1))):
```

Two factors appear repeatedly in these equations. One we denote as ePx, and one as eQx, as they are exponentials involving P and Q respectively. Making substitutions for these expressions reduce the number of variables in the equations.

```
> reds:=solve({ePx=1-RYp, eQx=1-RYs},{P,Q});
> adred1:=subs(reds,adeq1):
> adred2:=subs(reds,adeq2):
```

$$reds := \left\{ Q = -\frac{\ln(eQx) + \frac{kS\,Z}{\lambda} + \frac{kS\,FS}{\lambda} + \frac{kS\,FE0}{\lambda}}{kQ} \right.$$

$$\left. , P = -\frac{\ln(ePx) + kP\,FP}{kP} \right\}$$

Outlook™ must solve these adjoint equations to determine the optimal fertility levels. Since this must be done automatically, it is necessary to understand the structure of the equations and their solutions, in order that the computer algorithm can be robust. Having reduced the problem down to solving two non-linear equations ($adred1$, $adred2$) in two unknowns (ePx, eQx), Maple now assists in the simplification of these equations. The first step is to use the share

library utility "sprint" to examine the structure of the equations.

```
> read `c:\\maplev3\\share\\system\\sprint`;
> _EnvSprint:=200:
> sprint(normal(adred1));
> sprint(normal(adred2));
```

$$-<< +20 >> /((\beta - 1 + << *3 >> - << *4 >> - << *4 >> + << *5 >>)(-1 + ePx)\ kS\ eQx) = -<< +8 >>(\alpha - 1)/(\beta - 1 + << *3 >> - << *4 >> - << *4 >> + << *5 >>)(-1 + ePx)\ kS\ eQx\ \alpha)$$

$$-<< +66 >> /(g(\beta - 1 + << *3 >> - << *4 >> - << *4 >> + << *5 >>)(-1 + ePx)\ kS\ eQx) = -\frac{<< +46 >>(\alpha - 1)}{g << +6 >>(-1 + ePx)\ kS\ eQx\ \alpha}$$

The denominators on the left and right hand sides of the two equations differ only by a factor of α. The equations are thus significantly simplified by the stripping away the denominators. The equations are also converted into expressions.

```
> adredx1:=simplify(numer(op(1,adred1))
> *alpha-numer(op(2,adred1))):
> adredx2:=simplify(numer(op(1,adred2))
> *alpha-numer(op(2,adred2))):
```

Both of these expressions are quadratic in eQx.

```
> LHS:=solve(adredx1,eQx):
> whattype(LHS);
> RHS:=solve(adredx2,eQx):
> whattype(RHS);
```

exprseq
exprseq

We need to find the value of ePx that gives the same value of eQx from both quadratics.

Numerical example

The method of solution is best illustrated with an example. Since both ePx and eQx are exponentials of negative numbers, their values should lie in the range 0 to 1. Taking typical parameter values allows us to plot the relationships specified by *adredx1* and *adredx2* and identify which roots are the feasible ones. Here $LHS[1]$ and $RHS[2]$ are found to be the desired roots.

The parameter ε is not biologically defined, but is a Lagrange multiplier that allows the expenditure constraint to be satisfied. It is set initially to 0, in which case we are considering the problem where the farmer's fertilizer expenditure is unconstrained.

```
> params:={cP=3.0,beta=0.06,alpha=0.9,
> kP=0.01, r=300.0, a2=3.0, a1=3.0, cS=2.0,
>  kS=0.08, lambda=0.9, hP=0.004, Z=3.0,
> mu=0.07, kQ=0.05, g=4.0, Ps=3.0}:
> params:=params union {epsilon=0};
> plotsetup(plotoptions=`noborder`);
> plot(subs(params,{LHS[1],RHS[2]})
> ,ePx=0..1,0..1,labels=[`ePx`,`eQx`]);
```

$$params := \{g = 4.0, \varepsilon = 0, Ps = 3.0, \alpha = .9, kP = .01, cS = 2.0, a1 = 3.0, \lambda = .9, kS = .08, hP = .004, a2 = 3.0, Z = 3.0, \mu = .07, kQ = .05, \beta = .06, cP = 3.0, r = 300.0\}$$

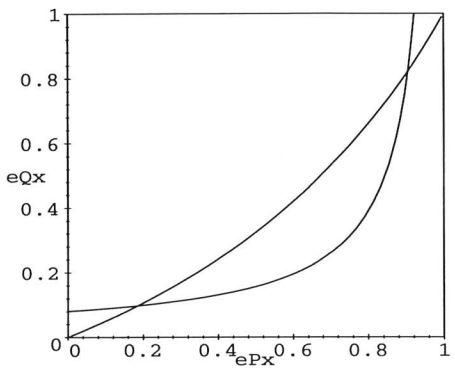


Figure 1: Curves along which the adjoint equations are satisfied.

The plot (Figure 1) shows two mathematically feasible points when both adjoint equations will be satisfied. However the right hand intersection corresponds to a negative value of *soilP* and *soilS*, which is biologically infeasible. Thus, it is the left hand solution which is of interest. This can be located numerically by finding the root of $f = adredx2 - adredx1$ which lies to the left of that function's minimum. Minimum and root-finding methods from Press *et al.* [4] were used in the software. For this example, the optimal solution lies to the left of 0.5:

```
> optePx:=fsolve(subs(params,LHS[1]-RHS[2])
> ,ePx,0..0.5):
> opteQx:=evalf(subs(ePx=optePx,subs(params
> ,LHS[1]))):
> solns:={ePx=optePx,eQx=opteQx};
```

$$solns := \{eQx = .09772983640, ePx = .1858351164\}$$

The optimal fertilizer application rates are then,

```
> solve(evalf(subs(params union solns,reds))
> ,{FP,FS});
```

$$\{FS = -.5624999999\,Q + 23.16241926 - 1.\,FE0$$

$$, FP = -1.\,P + 168.2895469\,\}$$

The optimal *soilP* and *soilS* levels are thus 168.3 and 23.2 respectively for this example. In the absence of a financial constraint, the optimal policy is to move to that level immediately in the first year and maintain it thereafter. In the software it is assumed that elemental sulphur is not used (i.e. $FE0 = 0$) since it is equivalent to an application of FS.

In the case where an expenditure constraint is required, the parameter ε must be adjusted iteratively until a value is found such that the cost is exactly equal to the financial constraint, d. At this point ε is equal to the marginal value of spending extra money. For example, if $\varepsilon = 0.7$ at this point, then for every extra dollar made available for fertilizer, NPV increases by \$0.70.

Computer code generation

The ability of Maple to convert expressions into C is very useful, because Outlook™ must calculate the values of $LHS[1]$ and $RHS[2]$ in its numerical solution routines, so that ePx can be adjusted until they coincide. The method used was to generate code for the quadratic coefficients of *adredx1* and *adredx2*, and the software then determined which root was needed. Code for these coefficients is easily generated, although the coefficient of eQx in *adredx2* benefits from some substitutions of variables — this shortens the resultant computer code that Maple generates. Furthermore, the following powers of ePx occur repeatedly and may be factored.

```
> rubble:={ePx^(hP/kP)=ePx1,ePx^((hP+kP)/kP)
> =ePx*ePx1,ePx^((2*kP+hP)/kP)=ePx^2*ePx1};

> adredx1:=subs(rubble,adredx1):

> adredx2:=subs(rubble,adredx2):
```

$$rubble := \left\{ ePx^{\left(\frac{2\,kP+hP}{kP}\right)} = ePx^2\ ePx1 \right.$$

$$\left. , ePx^{\left(\frac{hP+kP}{kP}\right)} = ePx\ ePx1, ePx^{\left(\frac{hP}{kP}\right)} = ePx1 \right\}$$

The expressions for the quadratic coefficients are:

```
> EA1:=factor(coeff(adredx1,eQx,2)):
> EB1:=factor(coeff(adredx1,eQx,1)):
> EC1:=factor(coeff(adredx1,eQx,0)):
> EA2:=factor(coeff(adredx2,eQx,2)):

> temp1:=coeff(adredx2,eQx,1):
> temp2:=expand(subs({epsilon=e1-1,mu=1-m1
> ,beta=1-b1},temp1)):

> EB2:=collect(temp2,[alpha,e1,m1]
> ,distributed,factor);
> EC2:=factor(coeff(adredx2,eQx,0)):
```

$$\begin{aligned} EB2 := (&-a1\ ePx\ ePx1\ cP\ kS - cS\ \lambda\ a1\ hP\ ePx1 \\ &-a1\ cP\ kS + cS\ \lambda\ a1\ hP\ ePx\ ePx1 \\ &+ a2\ cP\ kS\ ePx + a1\ ePx1\ cP\ kS \\ &-a2\ kP\ ePx\ cS\ \lambda + a1\ cP\ kS\ ePx \\ &+ a1\ ePx\ ePx1\ kP\ cS\ \lambda - a2\ cP\ kS \\ &-a1\ kP\ ePx\ cS\ \lambda)m1\ e1\ \alpha \\ &+ cS\ \lambda\ g\ kQ\ (\ ePx - 1\ )\ (\ a1\ hP\ ePx\ ePx1 \\ &-ePx1\ hP\ a1 + b1\ )\ e1\ \alpha \\ &- kS\ r\ (\ ePx - 1\ )\ (\ a1\ hP\ ePx\ ePx1 \\ &-ePx1\ hP\ a1 + b1\ )\ m1\ \alpha^2 + (-a2\ cP\ kS\ ePx \\ &+cS\ \lambda\ a1\ hP\ ePx1 + a1\ cP\ kS \\ &-a1\ ePx\ ePx1\ kP\ cS\ \lambda \\ &- a1\ cP\ kS\ ePx + a2\ cP\ kS + a2\ kP\ ePx\ cS\ \lambda \\ &+a1\ kP\ ePx\ cS\ \lambda + a1\ ePx\ ePx1\ cP\ kS \\ &-cS\ \lambda\ a1\ hP\ ePx\ ePx1 - a1\ ePx1\ cP\ kS)e1 \\ &+ kS\ r\ (\ ePx - 1\ )\ (\ a1\ hP\ ePx\ ePx1 \\ &-ePx1\ hP\ a1 + b1\ )\ \alpha \end{aligned}$$

These expressions are now coded for the computer programme.

```
> readlib(C):

> quads:=[qA1=EA1,qB1=EB1,qC1=EC1,qA2=EA2
> ,qB2=EB2,qC2=EC2]:

> C(quads,optimized);

> C(quads,filename=`quadcoes.c`,optimized);

    t1 = a1*cP;
    t5 = ePx-1.0;
    t6 = t5*t5;
    t7 = 1.0+epsilon;
    qA1 = -t1*kS*hP*alpha*ePx1*t6*t7;
    t11 = cP*kS;
    t13 = t11*alpha;
    t15 = ePx1*hP*a1;
    t18 = cS*lambda;
    t20 = a1*hP;
    t22 = t20*ePx*ePx1;
    t24 = alpha*cP;
    t27 = kS*ePx;
    t32 = ePx*ePx;
    qB1 = -t7*(t11*ePx-t11-t13*t15-kP*ePx
*t18+2.0*t13*t22-t24*kS*beta+t24*t27*beta
-t11*ePx*alpha+t13-t13*t32*ePx1*t20);
    t38 = kP*cS;
    qC1 = -t38*ePx*lambda*t7;
    t45 = alpha*mu;
    t46 = kS*r;
    t51 = t18*g*kQ;
    qA2 = -a1*alpha*hP*ePx1*t6*(t45*t46
-alpha*r*kS+t46+t51+epsilon*cS*lambda*g*kQ);
    t59 = a1*ePx;
    t62 = t59*ePx1*cP*kS;
    t63 = t18*t15;
    t64 = t1*kS;
    t68 = t18*a1*hP*ePx*ePx1;
    t69 = a2*cP;
    t70 = t69*t27;
    t72 = a1*ePx1*t11;
    t73 = a2*kP;
    t75 = ePx*cS*lambda;
    t76 = t73*t75;
    t77 = t1*t27;
    t80 = t59*ePx1*t38*lambda;
```

```
    t81 = t69*kS;
    t82 = a1*kP;
    t83 = t82*t75;
    t84 = -t62-t63-t64+t68+t70+t72-t76+t77
+t80-t81-t83;
    t86 = e1*alpha;
    t88 = t22-t15+b1;
    t89 = t5*t88;
    t94 = alpha*alpha;
    t97 = -t70+t63+t64-t80-t77+t81+t76+t83
+t62-t68-t72;
    qB2 = t84*m1*t86+t51*t89*t86-t46*t5*t88
*m1*t94+t97*e1+t46*t89*alpha;
    qC2 = -cS*t7*lambda*(-alpha+t45+1.0)
*(-t22-1.0-t59*ePx1*kP+t82*ePx+t73*ePx+t15
+beta);
```

This code was pasted into the numerical optimization portion of Outlook™. Starting with $\varepsilon = 0$, the solution (ePx, eQx) is located iteratively. This is then converted to an optimal nutrient level ($soilP$, $soilS$). The optimal fertilizer recommendation (FP, FS) is the difference between this level and the existing nutrient status (P, S). If this fertilizer recommendation can be implemented within the budgetary constraint, it is optimal. Otherwise the Lagrange multiplier ε is iteratively adjusted (using the bisection method) and the procedure repeated until the constraint is exactly satisfied. The recommendation at this point is then the constrained optimum. The whole process is repeated for each year, until the optimal policy for the 10 year time frame has been calculated.

Conclusion

This paper describes how a multi-dimensional optimal control problem was implemented in Maple. Maple provided a facility for simplifying and partially solving the problem, and for converting the less tractable equations into computer code so that they could be solved numerically.

The algorithm has performed well to date, and calculates fertilizer strategies which are economically superior to all other policies, although at times the improvement is small. When expenditure is unconstrained, the cost of fertilizer in the first year of the optimal strategy can be very high, as this is expected to be offset by high profit in future years. Experience has shown that imposing a constraint on expenditure can give a less risky policy — initial outlay is significantly reduced, and long term NPV is still near to that expected under the optimal unconstrained strategy [5].

Outlook™ runs under Windows and is available from AgResearch Soil Fertility Service, Ruakura Research Centre, Private Bag 3123, Hamilton, New Zealand, Tel: +64 7 838 5920, Fax: +64 7 838 5160.

References

[1] C.W. Clark: *Mathematical bioeconomics: the optimal management of renewable resources*, Wiley Interscience, New York, (1990).

[2] A.K. Metherell, D.G. McCall and S.J.R. Woodward: Outlook: a phosphorus fertilizer decision support model for grazed pastures, In: *Fertilizer requirements of grazed pasture and field crops* (L.D. Currie and P. Loganathan eds.), Occasional Report No. 8 of the Fertilizer and Lime Research Centre, Massey University, Palmerston North, New Zealand, pp. 24–39, (1995).

[3] A.K. Metherell, B.S. Thorrold and S.J.R. Woodward: Econometric modelling for fertilizer application, Proceedings of the 23rd Technical Conference of the New Zealand Fertilizer Manufacturers Research Association, November 1995, Rotorua, New Zealand, pp. 128–140, (1996).

[4] W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling: *Numerical recipes in Pascal: the art of scientific computing*, Cambridge University Press, New York, (1989).

[5] S.J.R. Woodward: A dynamic nutrient carryover model for pastoral soils and its application to optimizing fertilizer allocation to several blocks with a cost constraint, *Review of Marketing and Agricultural Economics*, **64**, pp. 75–85, (1996).

Biography

Simon Woodward is a farm systems modeler with the New Zealand Pastoral Agriculture Research Institute Ltd ("AgResearch"). He has a doctorate in applied mathematics and for the past few years has been using dynamical systems methodology to address questions in farm management, particularly the optimal grazing of pasture, and the optimal application of fertilizers.

Tips for Maple Instructors

Robert J. Lopez*

In this, the third column of the series, we again illustrate some Maple V Release 4 functionality of significance in the classroom. In particular, we examine the role technology has in shaping pedagogy. Inescapably, the available technology dictates the operable pedagogy. In the pencil-and-paper world from which we are emerging, the lack of graphics and convenient numerical calculations, for example, imposed an analytical style on didactics, overshadowing experimentation and investigation.

In this issue's column, we consider two major topics. First, we examine the traditional approach to finding the recursion relation for a power series solution of a differential equation and compare that approach to one found in [1], the *Maple V Flight Manual.* Then, we explore the optimizing property of the Fourier series. Each of these items reflects how the available technology influences the pedagogy. In particular, each requires replacing infinite sums with finite sums in Maple.

In a third section, we look at three Maple functionalities that affect the student's view of Maple in the classroom. First, we consider an error Maple makes in the midst of a lesson on removable singularities. Then, we contrast plotting a curve defined vectorially in two and three dimensions. Finally, we examine some of the difficulties branches pose for both students and Maple in the computation of curvature of a circle.

I continue to hope this column becomes a forum where common problems can be identified and resolved, where useful hints and strategems can be found, and where pedagogical insights can be shared. Hence, I urge all readers who have their own experiences with Maple in instruction, successes and failures alike, to communicate with me by e-mail (r.lopez@rose-hulman.edu), fax(812-877-3198), or letter (Math Dept., Rose-Hulman Institute of Technology, Terre Haute, IN 47803). As much as possible, I would like this column to address real issues, from real classes. Only the column's readers can make that happen.

Series solutions of ODEs

I recently succeeded in getting Maple to replicate the steps I learned 35 years ago for obtaining the recursion relation in a formal power series solution to an ordinary differential equation. In comparison to a simpler technique appearing in [1], I conclude that the cherished classical formalism is probably inferior to a streamlined Maple solution. I'll present both, and let you judge.

```
> with(student):
```

We write an arbitrary second-order linear equation with variable coefficients.

```
> q :=
> diff(y(x),x,x) + x^2*diff(y(x),x) + y(x)
> = 0;
```

$$q := \left(\frac{\partial^2}{\partial x^2}\,\mathrm{y}(x)\right) + x^2\left(\frac{\partial}{\partial x}\,\mathrm{y}(x)\right) + \mathrm{y}(x) = 0$$

Next, write a formal power series as the solution.

```
> Y :=sum(a[n]*x^n,n=0..infinity);
```

$$Y := \sum_{n=0}^{\infty} a_n\, x^n$$

Substitute the series into the differential equation.

```
> q1 := eval(subs(y(x)=Y,q));
```

$$q1 := \left(\sum_{n=0}^{\infty}\left(\frac{a_n\, x^n\, n^2}{x^2} - \frac{a_n\, x^n\, n}{x^2}\right)\right) + x^2\left(\sum_{n=0}^{\infty}\frac{a_n\, x^n\, n}{x}\right) + \left(\sum_{n=0}^{\infty} a_n\, x^n\right) = 0$$

All powers of x submit to a combination of **simplify** and **combine**.

```
> map(simplify@combine,q1);
```

$$\sum_{n=0}^{\infty}\Big(a_n\, x^{(n-2)}\, n^2 - a_n\, x^{(n-2)}\, n + x^{(n+1)}\, a_n\, n + a_n\, x^n\Big) = 0$$

However, the indices on each separate series on the left of *q1* must be shifted. We have to access and massage each such series separately. Unfortunately, this requires use of the "low-level" **op** command.

```
> for j from 1 to 3 do
> y.j := simplify(combine(op(j,lhs(q1))));
> od;
```

$$y1 := \sum_{n=0}^{\infty}\left(a_n\, x^{(n-2)}\, n^2 - a_n\, x^{(n-2)}\, n\right)$$

$$y2 := \sum_{n=0}^{\infty} x^{(n+1)}\, a_n\, n$$

*Department of Mathematics, Rose-Hulman Institute of Technology, Terre Haute, IN 47803 r.lopez@rose-hulman.edu

MapleTech Vol. 4, No. 3, pp. 52–59; ISSN 1061-5733

$$y3 := \sum_{n=0}^{\infty} a_n x^n$$

Shift indices so the power of x in the general term in each series is the same.

```
> y4 := changevar(n-2=k,y1,k);
> y5 := changevar(n+1=k,y2,k);
> y6 := changevar(n=k,y3,k);
```

$$y4 := \sum_{k=-2}^{\infty} (a_{2+k} x^k (2+k)^2 - a_{2+k} x^k (2+k))$$

$$y5 := \sum_{k=1}^{\infty} a_{k-1} x^k (k-1)$$

$$y6 := \sum_{k=0}^{\infty} a_k x^k$$

Next, each sum must start at the same value of the index k. Here, we must have $k = 1$. In the first sum, the terms corresponding to k = -2 and -1 are zero. In the first and third sums, the terms corresponding to $k = 0$ must be extracted separately. In Maple, however, it is difficult to manipulate the indices in a mathematically meaningful way. We settle for an artifice permitted by the **subs** command.

```
> y4a := subs(-2=1,y4);
> y6a := subs(0=1,y6);
```

$$y4a := \sum_{k=1}^{\infty} (a_{2+k} x^k (2+k)^2 - a_{2+k} x^k (2+k))$$

$$y6a := \sum_{k=1}^{\infty} a_k x^k$$

I suspect there would be a problem if "-2" or "-1" appeared in a context other than just the summation limits. Nonetheless, we are ready to combine the three transformed series.

```
> q2 := combine(y4a+y5+y6a);
```

$$q2 := \sum_{k=1}^{\infty} (a_{2+k} x^k (2+k)^2 - a_{2+k} x^k (2+k) + a_{k-1} x^k (k-1) + a_k x^k)$$

We want a single coefficient of x^k. Hence,

```
> q3 := factor(q2);
```

$$q3 := \sum_{k=1}^{\infty} x^k (2\, a_{2+k} + 3\, a_{2+k}\, k + a_{2+k}\, k^2 + a_{k-1}\, k - a_{k-1} + a_k)$$

The **coeff** command fails here, so we again resort to the low-level **op** command to access the summand of q3. (Fortunately, Release 5 will have a **summand** command to parallel the **integrand** command of the *student* package.)

```
> q4 := op(1,q3);
```

$$q4 := x^k (2\, a_{2+k} + 3\, a_{2+k}\, k + a_{2+k}\, k^2 + a_{k-1}\, k - a_{k-1} + a_k)$$

Now, **coeff** succeeds.

```
> q5 := coeff(q4,x^k);
```

$$q5 := 2\, a_{2+k} + 3\, a_{2+k}\, k + a_{2+k}\, k^2 + a_{k-1}\, k - a_{k-1} + a_k$$

Finally, we solve for a_{k+2} in terms of a_k and a_{k-1}, resulting in the sought-for recursion relation.

```
> isolate(q5,a[k+2]);
```

$$a_{2+k} = \frac{-a_{k-1}\, k + a_{k-1} - a_k}{2 + 3\, k + k^2}$$

Finally, with yet another artifice, we attend to the $k = 0$ terms in the series *y4* and *y6*.

```
> value(subs(infinity=0,y4+y6)) = 0;
```

$$2\, a_2 + a_0 = 0$$

Rather than continue with the rest of the classical manipulations, we implement a device from [1] wherein a finite sum replaces the infinite series Y. The secret is to make the terms in the finite sum generic, and to span across the general term, x^n. Thus,

```
> Q := sum(a[k]*x^k,k=n-1..n+2);
```

$$Q := a_{n-1} x^{(n-1)} + a_n x^n + a_{n+1} x^{(n+1)} + a_{2+n} x^{(2+n)}$$

Again, substitute into the differential equation.

```
> q6 := eval(subs(y(x)=Q,q));
```

$$\begin{aligned} q6 := {} & \frac{a_{n-1} x^{(n-1)} (n-1)^2}{x^2} \\ & - \frac{a_{n-1} x^{(n-1)} (n-1)}{x^2} + \frac{a_n x^n n^2}{x^2} - \frac{a_n x^n n}{x^2} \\ & + \frac{a_{n+1} x^{(n+1)} (n+1)^2}{x^2} - \frac{a_{n+1} x^{(n+1)} (n+1)}{x^2} \\ & + \frac{a_{2+n} x^{(2+n)} (2+n)^2}{x^2} - \frac{a_{2+n} x^{(2+n)} (2+n)}{x^2} \\ & + x^2 \left(\frac{a_{n-1} x^{(n-1)} (n-1)}{x} + \frac{a_n x^n n}{x} \right. \\ & \left. + \frac{a_{n+1} x^{(n+1)} (n+1)}{x} + \frac{a_{2+n} x^{(2+n)} (2+n)}{x} \right) \\ & + a_{n-1} x^{(n-1)} + a_n x^n + a_{n+1} x^{(n+1)} \\ & + a_{2+n} x^{(2+n)} = 0 \end{aligned}$$

When in doubt, **simplify**. This will tame all powers of x.

```
> q7 := simplify(q6);
```

$$q7 := a_n\,x^n + a_{n+1}\,x^{(n+1)} + a_{2+n}\,x^{(2+n)} + x^{(n+3)}\,a_{2+n}\,n + x^{(2+n)}\,a_{n+1}\,n + x^{(n+1)}\,a_n\,n + x^n\,a_{n-1}\,n - x^n\,a_{n-1} - 3\,a_{n-1}\,x^{(n-3)}\,n + 2\,a_{n-1}\,x^{(n-3)} + a_{n-1}\,x^{(n-3)}\,n^2 + a_{n-1}\,x^{(n-1)} + a_n\,x^{(n-2)}\,n^2 + 3\,a_{2+n}\,x^n\,n + a_{2+n}\,x^n\,n^2 + 2\,a_{2+n}\,x^n + a_{n+1}\,x^{(n-1)}\,n + a_{n+1}\,x^{(n-1)}\,n^2 - a_n\,x^{(n-2)}\,n + 2\,x^{(n+3)}\,a_{2+n} + x^{(2+n)}\,a_{n+1} = 0$$

Free of the baggage encumbering **Sum**, **coeff** now works. We **map** it onto both sides of the equation $q7$.

```
> q8 := map(coeff,q7,x^n);
```

$$q8 := a_n - a_{n-1} + 2\,a_{2+n} + a_{n-1}\,n + 3\,a_{2+n}\,n + a_{2+n}\,n^2 = 0$$

Finally, solving for a_{n+2}, we get, in agreement with our earlier result,

```
> isolate(q8,a[n+2]);
```

$$a_{2+n} = \frac{-a_{n-1}\,n - a_n + a_{n-1}}{2 + n^2 + 3\,n}$$

Optimum property of Fourier series

The discovery that for a function f(x) defined on [0, π], the coefficients of its Fourier sine series are given by

$$b_n = \tfrac{2}{\pi}\int_0^{\pi} \mathrm{f}(x)\sin(n\,x)\,dx$$

traditionally involves manipulating a formal infinite sum. In fact, these coefficients are those which minimize the integral

```
> q := Int((f(x) -
> Sum(s[n]*sin(n*x),n=1..infinity))^2,
>                          x=0..Pi);
```

$$q := \int_0^{\pi} (\mathrm{f}(x) - (\sum_{n=1}^{\infty} s_n \sin(n\,x)))^2\,dx$$

In Maple, however, the differentiation

```
> diff(q,s[n]);
```

$$0$$

fails, since Maple does not "see" the general term s_n in the data structure it uses to represent the infinite sum. Thus, the traditional pedagogy used here must change if it is to be implemented in Maple. Typically, this change amounts to working with an explicit finite sum as in

```
> q1 := Int((f(x) -
> sum(s[n]*sin(n*x),n=1..3))^2,x=0..Pi);
```

$$q1 := \int_0^{\pi} (\mathrm{f}(x) - s_1\sin(x) - s_2\sin(2\,x) - s_3\sin(3\,x))^2\,dx$$

Pay particular attention to the change from **Sum** to **sum**. Without that change the following differentiations fail.

```
> for k from 1 to 3 do
> eq.k := value(expand(diff(q1,s[k]),sin))
>         = 0;
> od;
```

$$eq1 := -2\int_0^{\pi} \sin(x)\,\mathrm{f}(x)\,dx + s_1\,\pi = 0$$

$$eq2 := -2\int_0^{\pi} \sin(2\,x)\,\mathrm{f}(x)\,dx + s_2\,\pi = 0$$

$$eq3 := -2\int_0^{\pi} \sin(3\,x)\,\mathrm{f}(x)\,dx + s_3\,\pi = 0$$

Solving each equation for its single Fourier coefficient yields

```
> for k from 1 to 3 do
> isolate(eq.k, s[k]);
> od;
```

$$s_1 = 2\,\frac{\int_0^{\pi} \sin(x)\,\mathrm{f}(x)\,dx}{\pi}$$

$$s_2 = 2\,\frac{\int_0^{\pi} \sin(2\,x)\,\mathrm{f}(x)\,dx}{\pi}$$

$$s_3 = 2\,\frac{\int_0^{\pi} \sin(3\,x)\,\mathrm{f}(x)\,dx}{\pi}$$

from which we generalize to the familiar result stated above.

Well, having to make this compromise left me feeling I had cheated my students. So, I thought I'd shift the same computations to a less familiar domain. I supposed a set of functions p0(x), p1(x), p2(x), ... with two properties:

1) $\int_{-1}^{1} p_n(x)\,p_m(x)\,dx = 0,\ n \neq m$

2) $\int_{-1}^{1} p_k(x)^2\,dx = \frac{2}{2\,k+1}$

I asked "Are these two properties enough to reproduce the minimization property of the Fourier series?"

To tell Maple that the functions p0(x), p1(x), p2(x) have these properties, define the following equations for Property (1).

```
> q01:=Int(p0(x)*p1(x),x=-1..1)=0;
> q02:=Int(p0(x)*p2(x),x=-1..1)=0;
> q12:=Int(p1(x)*p2(x),x=-1..1)=0;
```

$$q01 := \int_{-1}^{1} \mathrm{p0}(x)\,\mathrm{p1}(x)\,dx = 0$$

$$q02 := \int_{-1}^{1} \mathrm{p0}(x)\,\mathrm{p2}(x)\,dx = 0$$

$$q12 := \int_{-1}^{1} \mathrm{p1}(x)\,\mathrm{p2}(x)\,dx = 0$$

Then define the following equations for Property (2).

```
> q00:=Int(p0(x)^2,x=-1..1)=2;
> q11:=Int(p1(x)^2,x=-1..1)=2/3;
> q22:=Int(p2(x)^2,x=-1..1)=2/5;
```

$$q00 := \int_{-1}^{1} \mathrm{p0}(x)^2\,dx = 2$$

$$q11 := \int_{-1}^{1} \mathrm{p1}(x)^2\,dx = \frac{2}{3}$$

$$q22 := \int_{-1}^{1} \mathrm{p2}(x)^2\,dx = \frac{2}{5}$$

Now set up the same measure of performance as used for the Fourier series, namely, the integral of the square of the difference between the function f(x) and an approximating sum.

```
> Q :=
> Int((f(x)-s0*p0(x)-s1*p1(x)-s2*p2(x))^2,
>     x=-1..1);
```

$$Q := \int_{-1}^{1} (\mathrm{f}(x) - s0\,\mathrm{p0}(x) - s1\,\mathrm{p1}(x) - s2\,\mathrm{p2}(x))^2\,dx$$

Differentiate the measure of performance with respect to each of the three coefficients s0, s1, s2. Set the derivatives equal to zero to determine the values of the coefficients that minimize the measure of deviation Q.

```
> eq1 := diff(Q,s0) = 0;
> eq2 := diff(Q,s1) = 0;
> eq3 := diff(Q,s2) = 0;
```

$$eq1 := \int_{-1}^{1} -2(\mathrm{f}(x) - s0\mathrm{p0}(x) - s1\mathrm{p1}(x) - s2\mathrm{p2}(x))\mathrm{p0}(x)\,dx = 0$$

$$eq2 := \int_{-1}^{1} -2(\mathrm{f}(x) - s0\mathrm{p0}(x) - s1\mathrm{p1}(x) - s2\mathrm{p2}(x))\mathrm{p1}(x)\,dx = 0$$

$$eq3 := \int_{-1}^{1} -2(\mathrm{f}(x) - s0\mathrm{p0}(x) - s1\mathrm{p1}(x) - s2\mathrm{p2}(x))\mathrm{p2}(x)\,dx = 0$$

Progress solving these equations depends on simplifying them. The parentheses need to be multiplied out, and any possible integrations done.

```
> eq4 := expand(eq1);
> eq5 := expand(eq2);
> eq6 := expand(eq3);
```

$$eq4 := -2\int_{-1}^{1} \mathrm{p0}(x)\,\mathrm{f}(x)\,dx + 2\,s0\int_{-1}^{1} \mathrm{p0}(x)^2\,dx + 2\,s1\int_{-1}^{1} \mathrm{p0}(x)\,\mathrm{p1}(x)\,dx + 2\,s2\int_{-1}^{1} \mathrm{p0}(x)\,\mathrm{p2}(x)\,dx = 0$$

$$eq5 := -2\int_{-1}^{1} \mathrm{p1}(x)\,\mathrm{f}(x)\,dx + 2\,s0\int_{-1}^{1} \mathrm{p0}(x)\,\mathrm{p1}(x)\,dx + 2\,s1\int_{-1}^{1} \mathrm{p1}(x)^2\,dx + 2\,s2\int_{-1}^{1} \mathrm{p1}(x)\,\mathrm{p2}(x)\,dx = 0$$

$$eq6 := -2\int_{-1}^{1} \mathrm{p2}(x)\,\mathrm{f}(x)\,dx + 2\,s0\int_{-1}^{1} \mathrm{p0}(x)\,\mathrm{p2}(x)\,dx + 2\,s1\int_{-1}^{1} \mathrm{p1}(x)\,\mathrm{p2}(x)\,dx + 2\,s2\int_{-1}^{1} \mathrm{p2}(x)^2\,dx = 0$$

No integrations have been done because everything is symbolic. Maple knows only properties (1) and (2) apply, but can't apply them until told. Do this with a **simplify** command containing the equations that define Property (1) and Property (2).

```
> q0 :=
> simplify(eq4,{q00,q11,q22,q01,q02,q12});

> q1 :=
> simplify(eq5,{q00,q11,q22,q01,q02,q12});

> q2 :=
> simplify(eq6,{q00,q11,q22,q01,q02,q12});
```

$$q0 := -2\int_{-1}^{1} \mathrm{p0}(x)\,\mathrm{f}(x)\,dx + 4\,s0 = 0$$

$$q1 := -2\int_{-1}^{1} \mathrm{p1}(x)\,\mathrm{f}(x)\,dx + \frac{4}{3}\,s1 = 0$$

$$q2 := -2\int_{-1}^{1} \mathrm{p2}(x)\,\mathrm{f}(x)\,dx + \frac{4}{5}\,s2 = 0$$

Solve each equation for the one coefficient it contains.

```
> isolate(q0,s0);
> isolate(q1,s1);
> isolate(q2,s2);
```

$$s0 = \frac{1}{2}\int_{-1}^{1} \mathrm{p0}(x)\,\mathrm{f}(x)\,dx$$

$$s1 = \frac{3}{2}\int_{-1}^{1} \mathrm{p1}(x)\,\mathrm{f}(x)\,dx$$

$$s2 = \frac{5}{2}\int_{-1}^{1} \mathrm{p2}(x)\,\mathrm{f}(x)\,dx$$

Generalize these definitions to a formula for the nth coefficient s_n.

$$s_n = \left[\tfrac{2k+1}{2}\right] \int_{-1}^{1} \mathrm{f}(x)\,p_n(x)\,dx$$

Now, it was time to ask if such functions exist. Are there actually functions with properties (1) and (2)? My students didn't know, be we surely do, that the the functions $p_n(x)$ are the Legendre polynomials, found in the *orthopoly* package.

```
> with(orthopoly);
```

$$[G,\ H,\ L,\ P,\ T,\ U]$$

A convenient way to access the first five Legendre polynomials uses the loop

```
> for k from 0 to 4 do
> p.k:=P(k,x);
> od;
```

$$p0 := 1$$

$$p1 := x$$

$$p2 := \frac{3}{2}x^2 - \frac{1}{2}$$

$$p3 := \frac{5}{2}x^3 - \frac{3}{2}x$$

$$p4 := \frac{35}{8}x^4 - \frac{15}{4}x^2 + \frac{3}{8}$$

Consider the function $f(x) = \sin(\pi x)$ on the interval [-1,1], and compute the coefficients s0, s1, s2, s3, s4 by the integral formulas derived above.

```
> for k from 0 to 4 do
> A[k]:=(k+1/2)*int(sin(Pi*x)*p.k,x=-1..1);
> od;
```

$$A_0 := 0$$

$$A_1 := \frac{3}{\pi}$$

$$A_2 := 0$$

$$A_3 := 7\,\frac{\pi^2 - 15}{\pi^3}$$

$$A_4 := 0$$

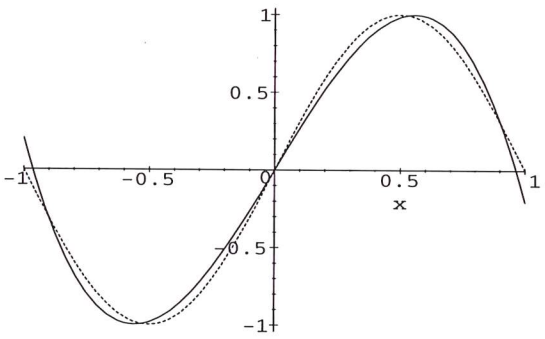
Form an approximating sum with these five coefficients. Note how the new **add** command lets us avoid the quotes needed by **sum**.

```
> s := add(A[n]*p.n,n=0..4);
```

$$s := 3\,\frac{x}{\pi} + 7\,\frac{(\pi^2 - 15)\left(\frac{5}{2}x^3 - \frac{3}{2}x\right)}{\pi^3}$$

Plot the approximation and the function f(x) on the same set of axes.

```
> plot([sin(Pi*x),s],
> x=-1..1,linestyle=[2,1],
> color=black);
```



Miscellaneous Maple observations

In this concluding section we consider three Maple characteristics of consequence for students whose mathematical instincts are still in the formative stages. First, we consider an item from a discussion of removable singularities. Then we consider plotting curves defined in vector notation. Finally, we look at the subtleties of branches in the calculation of the curvature of a circle.

EVALUATE-AT

The function

```
> f := sin(3*x)/sin(4*x);
```

$$f := \frac{\sin(3\,x)}{\sin(4\,x)}$$

is easily found to have a removable discontinuity at $x = 0$. Maple correctly evaluates

```
> q := Limit(f,x=0);
```

$$q := \lim_{x \to 0} \frac{\sin(3\,x)}{\sin(4\,x)}$$

as

```
> value(q);
```

$$\frac{3}{4}$$

In an effort to show that there is indeed a discontinuity at $x = 0$ we try

```
> subs(x=0,f);
```

$$1$$

a terribly erroneous result that steals all thunder from a lesson on removable discontinuities. Clearly, Maple has calculated $\frac{\sin(0)}{\sin(0)}$ and simplified that fraction to 1. The "error" is that Maple did not evaluate both numerator and denominator to 0. However, observe that if the expression f had been entered as a *function*, Maple's behavior would be different.

```
> F := unapply(f,x);
```

$$F := x \to \frac{\sin(3\,x)}{\sin(4\,x)}$$

```
> F(0);

Error, (in F) division by zero
```

So, we observe, there is a difference between evaluating a function and substituting into an expression. We see this difference again arising in the case of piecewise-defined functions.

```
> g := piecewise(x<0,x, x<2,x^2);
```

$$g := \begin{cases} x & x < 0 \\ x^2 & x < 2 \end{cases}$$

```
> subs(x=1,g);
```

$$\begin{cases} 1 & 1 < 0 \\ 1 & 1 < 2 \end{cases}$$

Substitution does not behave "mathematically" as "plug in and evaluate." However, if g is redefined as a *function*, evaluation occurs as expected.

```
> G := unapply(g,x);
```

$$G := x \to \text{piecewise}(x < 0,\ x,\ x < 2,\ x^2)$$

```
> G(1);
```

$$1$$

Mike Monagan alerts me to a new paradigm coming in Release 5. Since **subs** is not the exact mathematical equivalent of *substitution*, a new **eval** command will support the notion of "plug in and evaluate at." Both of the examples above yield to this new functionality.

PLOTTING CURVES IN VECTOR FORM

The vector representation of a curve is a staple of a multivariable calculus course. Hence, representing a helix via the radius vector **R** = cos(t) **i** + sin(t) **j** + t **k** is most easily done in Maple with the syntax

```
> with(linalg):
> with(plots):
> R := vector([cos(t),sin(t),t]);
```

$$R := \begin{bmatrix} \cos(t) \\ \sin(t) \\ t \end{bmatrix}$$

A plot of the helix is simply obtained via the syntax **spacecurve**(R, t = 0..4*Pi) but the analogous syntax for a plane curve will fail. For example, defining the plane curve

```
> r := vector([cos(t),sin(t)]);
```

$$r := \begin{bmatrix} \cos(t) \\ \sin(t) \end{bmatrix}$$

and using the syntax **plot**(r, t = 0..2*Pi), fails to yield the expected circle since Maple treats the *vector* **r** as a *list* of two functions and plots a sine and a cosine curve. We are forced to treat the curve parametrically and address the components of the vector **r**. Hence, plotting a plane curve defined as a vector **r** requires **plot**([r[1], r[2], t = 0..2*Pi]). Such blurring of the roles of the *vector* and *list* data-structures in Maple requires students to remember syntactical particulars that add to the instructional overhead.

CURVATURE OF A CIRCLE

Another staple in a multivariable calculus course is the notion of curvature of a plane curve. In fact, a standard shake-down of the definition of curvature is the verification that the curvature of a circle is a constant. This calculuation, however, requires deft handling of branches of the square root funtion.

We illustrate with a circle centered at the origin.

```
> q := x^2 + y^2 = a^2;
```

$$q := x^2 + y^2 = a^2$$

First, we obtain $y(x)$ explicitly, and compute the curvature on the upper and lower semicircles separately.

```
> qq := solve(q,y):
> y1 := qq[1];
> y2 := qq[2];
```

$$y1 := \sqrt{-x^2 + a^2}$$

$$y2 := -\sqrt{-x^2 + a^2}$$

Defining curvature as $\kappa = \frac{y''}{[1+(y')^2]^{(\frac{3}{2})}}$ we get

```
> kappa[1] :=
> diff(y1,x,x)/(1+diff(y1,x)^2)^(3/2);
> kappa[2] :=
> diff(y2,x,x)/(1+diff(y2,x)^2)^(3/2);
```

$$\kappa_1 := \frac{-\frac{x^2}{(-x^2+a^2)^{3/2}} - \frac{1}{\sqrt{-x^2+a^2}}}{(1+\frac{x^2}{-x^2+a^2})^{3/2}}$$

$$\kappa_2 := \frac{\frac{x^2}{(-x^2+a^2)^{3/2}} + \frac{1}{\sqrt{-x^2+a^2}}}{(1+\frac{x^2}{-x^2+a^2})^{3/2}}$$

If we use simplification with respect to the side relation q, and a **radsimp**, we obtain

```
> radsimp(simplify(kappa[1],{q}));
> radsimp(simplify(kappa[2],{q}));
```

$$-\frac{1}{a}$$

$$\frac{1}{a}$$

The difference in signs is significant. In contrast to newer calculus texts which define curvature with an absolute value, older calculus texts such as [2], define curvature so as to preserve information about concavity contained in the sign of y//. Hence, on the upper semicircle, $y(x)$ is concave downwards so y// is negative, whereas the opposite is true on the lower semicircle. Maple has yielded the correct results!

But we have violated Monagan's Prime Directive: *Thou shalt not use* **radsimp** *under any circumstances! That command exists for backwards compatibility only. Thou shalt use* **simplify**. It was luck that got us the correct answers. The weakness of **radsimp**, and the correct way to deal with the branches, are illustrated by the following computation based on implicit differentiation as implemented by the **implicitdiff** command.

```
> yx := implicitdiff(q,y,x);
> yxx := implicitdiff(q,y,x,x);
```

$$yx := -\frac{x}{y}$$

$$yxx := -\frac{x^2+y^2}{y^3}$$

```
> kappa := yxx/(1+yx^2)^(3/2);
```

$$\kappa := -\frac{x^2+y^2}{y^3\,(1+\frac{x^2}{y^2})^{3/2}}$$

If we use the same combination of **simplify** and **radsimp** as before, we get

```
> radsimp(simplify(kappa,{q}));
```

$$-\frac{1}{a}$$

Where exactly did we lose the sign information inherent in the curvature of the branches?

Clearly, simplifying $[y^2]^{(\frac{3}{2})}$ to y^3 is the culprit. It should be $|y|^3$ so that $[y/|y|]^3$ is either 1 or -1, depending on the branch y represents. The mathematically correct way (and the proper Maple way) to proceed is to simplify κ with respect to the defining circle, then make assumptions on the signs of a, and y.

```
> q1 := simplify(kappa,{q});
```

$$q1 := -\frac{a^2}{y^3\,(\frac{a^2}{y^2})^{3/2}}$$

```
> assume(a>0);
> interface(showassumed=0);
```

(Incidentally, my students discovered before I did that the above **interface** command builds into the worksheet the suppression of the tilde on assumed variables. The alternative is to remember during the lecture to use the Options menu. And it is puzzling that the suppression of the tilde is not permanent. Saving the worksheet and re-opening it later finds the tildes have reappeared.)

Returning to the computation at hand, on the upper semicircle, $y(x) > 0$, so

```
> assume(y>0);
> simplify(q1);
```

$$-\frac{1}{a}$$

whereas on the lower semicircle, $y(x) < 0$, so

```
> assume(y<0);
> simplify(q1);
```

$$\frac{1}{a}$$

References

[1] Wade Ellis, Eugene Johnson, Ed Lodi, and Dan Schwalbe: *Maple V flight manual*, Brooks/Cole, (1992).

[2] William Anthony Granville, Percey F. Smith, and William Raymond Longley: *Elements of the differential and integral calculus*, Ginn and Company, (1934).

Biography

Robert J. Lopez is a classically trained applied mathematician with a Purdue University Ph.D. (1970) in Relativistic Cosmology. After a short stint at the University of Nebraska-Lincoln, he spent 12 years at Memorial University in St. John's, Newfoundland, Canada, an odyssey of cod fish, ice hockey, and long gray winters. At Rose-Hulman Institute of Technology since 1985 where he pioneered the use of Maple in the classroom, he has authored books and papers, represented Maple "on the road" for 30 months, and received his Institute's awards for both teaching excellence and distinguished scholarship. He continues to promote technology as an active partner in undergraduate instruction and curriculum revision.

Exploring Population Dynamics with Maple

Walter Middleton and Douglas Curran *

Abstract: The paper outlines the general philosophy of the approach used to introduce discrete-system modeling to undergraduates at the University of Sunderland who are registered in a wide variety of courses including education, business and the sciences. A Leslie population model is developed and the underlying assumptions of its implementation considered. The model is then run and the results discussed. The work is presented in a form suitable for use by other university instructors.

Introduction

The School of Computing and Information Systems at the University of Sunderland offers modules in mathematical modeling principally aimed at students registered in the University Combined Programme (UCP) and the BA(Accounting and Mathematics) (BAAM) degrees. In the summer of 1992 the BAAM degree was validated and at the same time the mathematics offered to UCP students was entirely re-written with the specific aim of including a computer algebra system as a core, essential part of the resources necessary to enable students to explore, learn and do mathematics. The system chosen was Maple. As well as using Maple as a teaching and learning tool for topics such as the calculus and linear algebra offered at a variety of levels, staff have integrated Maple into the modeling courses offered. This presents the students with a unified approach to both the initial teaching and the practical application of mathematics. Although the application considered in this paper is purely numerical, Maple is used by students who employ its symbolic capabilities throughout their studies. It is anticipated that the experience gained at Sunderland by integrating a computer algebra system (rather than using it as an add-on) into the teaching and learning process across a range of related modules will be of use to other university staff involved in or contemplating similar activities.

Foundation courses in modeling at Sunderland

After taking first year courses aimed at giving them an acceptable minimum background in the calculus, linear algebra and statistics, second year students are offered a one semester module titled "The Foundations of Mathematical Modeling". Typically, second year classes in modeling are comprised of students from a variety of disciplines including science, accountancy and education.

Our second year modeling courses are conventional in the sense that they encourage teams of students to look at a problem from several viewpoints and use simplifying assumptions and techniques such as developing feature lists describing the essentials of a problem before attempting to build a mathematical model. Once this is successfully done, the students approach the problem of solving a system of equations and interpreting the solution obtained in terms of the original model. It is at this stage of the modeling process that Maple may be used to advantage. The underlying philosophy that the solution depends on the assumptions is stressed throughout and students are positively encouraged to re-define their assumptions and refine their models to obtain solutions which describe the original scenario ever more closely.

Exploring the solution stage of the modeling process (by varying parameters and re-solving) is essential if students are to obtain a "feel" for the work they are doing and it is at this stage that computer algebra systems can offer much to practitioners and students alike. We shall now consider a typical modeling scenario in which students benefit by employing Maple as a solution tool.

Population modeling

Since 1991 the authors have been involved in TEMPUS funded projects with Bulgaria and Hungary. From 1991–94 a project titled "New Approaches to the Teaching of Engineering Mathematics" involved the authors working with staff of the Technical University of Sofia, Bulgaria and from 1994 to the present time the authors have worked with staff from the Budapest University of Economic Sciences, Budapest, Hungary on a project titled "The Development and Teaching of Modeling Courses in Information-Based Decision Management." Both projects have used the vehicle of population modeling to teach students the basic concepts of the modeling process and both projects have faced similar situations when implementing modeling courses. Commonly, we have found:

i) classes comprising students who are theoretically capable and well-schooled in the traditional skills of mathematics but who are inexperienced applied mathematicians;

ii) students who have little or no previous experience of modeling;

iii) at second year level, students have little real experience of working in teams and many of them find dif-

*University of Sunderland United Kingdom

MapleTech Vol. 4, No. 3, pp. 60–64; ISSN 1061-5733

ficulty with the concept that the outcome of a problem scenario depends on the assumptions made to build a mathematical model and that there are no clear-cut right and wrong answers;

iv) students who have no experience in the essential role that computers now play in the analysis and exploration of mathematical models;

v) students who have no experience of working in a mathematics laboratory consisting of microcomputers equipped with suitable software;

vi) students whose assessment regimes consist (almost) entirely of formal examinations.

Modeling the population of the blue whale using Maple

Following introductory work in which the students consider continuous time-variable exponential and logistic population models, the scenario described below is introduced as the first serious example of a model using a discrete time-variable. One of the best known discrete model formulations is that proposed by P. H. Leslie [1]. The Leslie model predicts the age structure of a population of animals after a unit period of time has passed given that the structure at the start of the time period and the appropriate information regarding mortality and fertility rates is known. Following the work of Leslie, the development of a typical initial model, together with some simplifying assumptions is outlined below.

Firstly, since only the females in a population can give birth, the model considers the female population only. Secondly, we break the female population up into $m+1$ age groups denoted by $0, 1, 2, 3, \cdots, m$. The number of females in age group x at time t is denoted by $n_{x,t}$.

MORTALITY RATES

The proportion of females surviving group x to become members of group $x+1$ at time $t+1$ is denoted by P_x. In general, $P_x \geq 0$ but $P_m = 0$, since the proportion surviving the last age group must be zero by definition.

FERTILITY RATES

The average number of female calves born per member of group x at time t is denoted by F_x. We assume that all of these calves survive to become members of group 0 at time $t+1$. Further, we assume that the mortality and fertility rates stay constant over several time periods. The above assumptions and notation are used to build the initial model outlined below.

Relationships are first established giving the number of females in each age group at time $t+1$. Taking the lowest age group first, the number of members in this group is given by the formula:

$$n_{0,t+1} = F_0 n_{0,t} + F_1 n_{1,t} + F_2 n_{2,t} + \cdots + F_m n_{m,t}$$

The numbers in the remaining classes follow a clear pattern which may be represented by the relationships:

$$n_{1,t+1} = P_0 n_{0,t}$$

$$n_{2,t+1} = P_1 n_{1,t}$$

$$\vdots$$

$$n_{m,t+1} = P_{m-1} n_{m-1,t}$$

The above relationships are best expressed in matrix form as shown below.

$$\begin{bmatrix} n_{0,t+1} \\ n_{1,t+1} \\ n_{2,t+1} \\ n_{3,t+1} \\ \vdots \\ n_{m,t+1} \end{bmatrix} =$$

$$\begin{bmatrix} F_0 & F_1 & F_2 & \cdots & F_{m-1} & F_m \\ P_0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & P_1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & P_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & P_{m-1} & 0 \end{bmatrix} \begin{bmatrix} n_{0,t} \\ n_{1,t} \\ n_{2,t} \\ n_{3,t} \\ \vdots \\ n_{m,t} \end{bmatrix}$$

The derivation of this model is a challenge for most of our students presenting, as it does, a considerable increase in conceptual difficulty when compared with the exponential growth and logistic growth models described by simple differential equations.

The above matrix equation is easily represented as

$$\mathbf{n}_{t+1} = \mathbf{M}\mathbf{n}_t$$

from which it is clear that

$$\mathbf{M}_1 = \mathbf{M}\mathbf{n}_0$$

$$\mathbf{M}_2 = \mathbf{M}\mathbf{n}_1 = \mathbf{M}^2\mathbf{n}_0$$

$$\mathbf{M}_3 = \mathbf{M}\mathbf{n}_2 = \mathbf{M}^3\mathbf{n}_0$$

$$\vdots$$

so that in general, after p time periods, the population may be described by the matrix equation

$$\mathbf{M}_p = \mathbf{M}^p\mathbf{n}_0$$

The matrix $\mathbf{M}$ is square, of order $m+1$ and has interesting mathematical properties which students may be asked to investigate with advantage. However, in terms of the modeling process, we need to note that there are $m+1$ eigenvalues λ and corresponding eigenvectors $\mathbf{x}$ which satisfy

$$\mathbf{Mx} = \lambda\mathbf{x}$$

and that we can use certain properties of the matrix $\mathbf{M}$ and interpret them in terms of our population model.

The properties are:

i) there exists a unique positive value of λ say λ_m with the property that the corresponding eigenvector $\mathbf{x}_0$ can be thought of as having only non-negative elements;

ii) λ_m has the greatest absolute value of any eigenvalue in the system.

The theory supporting these properties is explained in detail to undergraduate students. While we have found that postgraduate students are capable of deriving the properties as a set piece of work, this is not normally the case with our undergraduates.

From a modeling point of view, property i) tells us that the Leslie model will always give a population distribution consisting of positive numbers only. This is important since it is impossible to consider populations with negative or complex numbers of animals! The eigenvector $\mathbf{x}_0$ corresponding to λ_m gives the steady-state age distribution of the population.

Property i) also states that the age distribution is unique since λ_m is unique. Property ii) tells us that we can easily find the value of λ_m by finding the eigenvalue with the greatest absolute value.

Maple will easily find the complete eigensystem of a matrix so that it is a simple matter to pick out the value of the eigenvalue λ_m and the corresponding eigenvector $\mathbf{x}_0$.

Model data

The following information has been obtained from studies of the blue whale [2] and enables us, following the mathematics above, to build a Leslie model of the population.

1) Female blue whales reach maturity at between four and seven years of age;

2) the gestation period for a pregnant female is about one year;

3) a single calf is born and requires nursing by the mother for about seven months;

4) nursing mothers do not become pregnant;

5) observations show that in general not more than one calf is born to a female every two years;

6) the observed male:female ratio in actual catches of blue whales is roughly 1:1;

7) the rate of breeding for older whales is slightly less than that of whales in their prime;

8) natural mortality appears to account for the deaths of about 13% of the whales in any given group except the final group where the mortality must be 100%;

9) the maximum age of blue whales is about 40 years.

The following table gives the estimated fertility and mortality rates for the given age distribution of the animal.

Age of Whale	0-	2-	4-	6-	8-
Fertility Rate[1]	0.00	0.00	0.19	0.44	0.50
Mortality Rate[2]	13	13	13	13	13
Age of Whale	10-	12-	14-	16-	18-
Fertility Rate[1]	0.50	0.45	0.45	0.45	0.45
Mortality Rate[2]	13	13	13	13	13
Age of Whale	20-	22-	24-	26-	28-
Fertility Rate[1]	0.45	0.45	0.45	0.45	0.45
Mortality Rate[2]	13	13	13	13	13
Age of Whale	30-	32-	34-	36-	38-
Fertility Rate[1]	0.45	0.45	0.45	0.45	0.00
Mortality Rate[2]	13	13	13	13	100

[1] Calves/female born in a two-year period

[2] Percentage of deaths in each age group

The information in the table gives rise to a Leslie matrix of order 20 which is used in the first modeling scenario considered below. Essentially the whales are assumed to breed at a roughly constant rate over their later adult life and have a constant survival rate over their whole life except when they reach the 38-40 class. The data indicate that no whale survives this age group.

Exploring the model

The essential Maple commands used are summarized below.

```
> with(linalg);
> P:=([seq(x,i=0..n)]);
> vals:=evalf(Eigenvals(A,vecs),4);
> x:=evalf(col(vecs,a),4);
> x:=evalf(evalm(x/x[1]*1000),4);
> map(round,x);
```

MODEL 1

```
> with(linalg):
> P1:=[0.87$18,0];
```

$$P1 := [.87, .87, .87, .87, .87, .87, .87, .87, .87, .87, .87, .87, .87, .87, .87, .87, .87, .87, 0]$$

```
> F1:=[0,0,0.19,0.44,0.5,0.5,0.45$13,0];
```

$$F1 := [0, 0, .19, .44, .5, .5, .45, .45, .45, .45, .45, .45, .45, .45, .45, .45, .45, .45, 0]$$

```
> L1:=matrix(20,20,0):
> for i to 20 do L1[1,i]:=F1[i] od:
> for i to 19 do L1[1+i,i]:=P1[i] od:
```

We will now find the eigensystem of L1.

```
> sys1:=evalf(Eigenvals(L1,vecs1),4);
```

$$sys1 := [1.111, .7807 + .3187\,I, .7807 - .3187\,I, .6132 + .5614\,I, .6132 - .5614\,I, .4004 + .7406\,I, .4004 - .7406\,I, .1676 + .8317\,I, .1676 - .8317\,I, -.09145 + .8219\,I, -.09145 - .8219\,I, -.3460 + .7418\,I, -.3460 - .7418\,I, -.5609 + .5885\,I, -.5609 - .5885\,I, -.8019 + .1281\,I, -.8019 - .1281\,I, -.7173 + .3762\,I, -.7173 - .3762\,I, 0]$$

Pick out the eigenvector corresponding to the steady-state

```
> x:=evalf(col(vecs1,1),4);
```

$$x := [-.6239, -.4881, -.3823, -.2986, -.2327, -.1814, -.1410, -.1092, -.0854, -.0670, -.05228, -.04122, -.03273, -.02603, -.02092, -.01697, -.01396, -.01154, -.009399, 0]$$

and normalize the population distribution to give a population of 1000 for age class 0-2.

```
> x:=evalf(evalm(x/x[1]*1000),4);
```

$$x := [1000., 782.4, 612.8, 478.7, 373.0, 290.8, 226.0, 175.0, 136.9, 107.4, 83.80, 66.08, 52.47, 41.73, 33.53, 27.20, 22.38, 18.50, 15.07, 0]$$

The rounded population distribution of the blue whale predicted from the first model is given by:

```
> PopModel1:=map(round,x);
```

$$PopModel1 := [1000, 782, 613, 479, 373, 291, 226, 175, 137, 107, 84, 66, 52, 42, 34, 27, 22, 19, 15, 0]$$

MODEL 2

The second model assumes that the fertility and mortality rates used follow the same pattern as those in Model 1 with respect to the initial age ranges, but after 14 years of age (about one third of their natural lives) the fertility rates are assumed to fall in a linear fashion with animals in the last two classes, like those in the first two, not breeding. The survival rates also follow the same pattern as those in Model 1 with respect to animals in the initial classes, but are again assumed to fall linearly after the main breeding period of their life has ended.

The input of the matrix L2 follows the same pattern as the input of matrix L1; it is in fact, a modified version of this matrix.

```
> P2:=[.87, .87, .87, .87, .87, .87, .87,
> .798, .725, .652, .580, .507, .435, .362,
> .290, .217, .145, .072, 0]:

> F2 := [0, 0, .19, .44, .5, .5, .5, .458,
> .417, .375, .333, .291, .250, .208, .166,
> .125, .083, .041, 0, 0]:

> L2 := matrix(20,20,0):
> for i to 20 do L2[1,i]:=F2[i] od:
> for i to 19 do L2[1+i,i]:=P2[i] od:

> sys2:=evalf(Eigenvals(L2,vecs2),4);
```

$$sys2 := [0, 1.088, .5231 + .4828\,I, .5231 - .4828\,I, .2118 + .5425\,I, .2118 - .5425\,I, .09085 + .5978\,I, .09085 - .5978\,I, -.04870 + .4359\,I, -.04870 - .4359\,I, -.4170 + .1716\,I, -.4170 - .1716\,I, -.4213, -.3101 + .2934\,I, -.3101 - .2934\,I, -.1472 + .3870\,I, -.1472 - .3870\,I, -.2346 + .1249\,I, -.2346 - .1249\,I, 0]$$

```
> x:=evalf(col(vecs2,2),4);
```

$$x := [-.7588, -.6059, -.4860, -.3889, -.3117, -.2495, -.1995, -.1601, -.1179, -.0788, -.04716, -.02519, -.01176, -.004658, -.001522, -.0003969, -.0000788, -.00001099, -.7279\,10^{-6}, 0]$$

The rounded population distribution of the blue whale predicted from the second model is given by:

```
> PopModel2:=map(round,evalm(x/x[1]*1000));
```

$$PopModel2 := [1000, 798, 640, 513, 411, 329, 263, 211, 155, 104, 62, 33, 15, 6, 2, 1, 0, 0, 0, 0]$$

The second model predicts that blue whales reach a maximum age of about 32 years. It is known from the given data that they reach a maximum age of about 40 years. This suggests that the assumptions made regarding the fertility and mortality rates should be revised, perhaps allowing a higher rate of breeding, possibly for longer and/or a higher rate of survival, possibly for longer.

Once this model is well understood, Sunderland undergraduates then study a Leslie model of the population of Red Deer on the Isle of Rhum, Scotland [2]. This model represents a further increase in difficulty for undergraduates in that the fertility and mortality rates of the animals have to be calculated from original data before the Leslie matrix can be written down. Sunderland postgraduates also study this model but are expected to be much more independent of the lecturer in charge of the class.

Conclusion

Once the foundations of the model are well understood, it is relatively easy to develop and explore other related scenarios using Maple to do the calculations. Such calculations are tedious to say the least if done by hand and can easily act as a barrier to students wishing to revise their assumptions and explore a modeling scenario. Once the normalized populations are calculated the results may be compared in a critical manner relating the model predictions to the original data and the assumptions used to build the model. Each model explored can lead to a different, more refined model, as the underlying assumptions are developed with experience.

The authors have sought to illustrate that it is possible to use Maple to explore certain types of modeling scenarios met typically in the life sciences and that the package can be used to advantage by students who are not mathematics majors and do not have a substantial knowledge of Maple.

References

[1] P.H. Leslie: On the use of matrices in certain population mathematics, *Biometrika*, **33**, pp.183–212, (1945).

[2] N.B. Usher: *Developments in the Leslie Matrix Model*, Department of Biology, University of York, U.K. .

Biographies

Walter Middleton obtained his BSc degree from the University of Hull in 1963 and his MSc from the CNAA in 1978. Since 1975 he has been a Senior Lecturer at the University of Sunderland (formerly Sunderland Polytechnic) and presently holds the post of Reader in Mathematics Learning Systems in the School of Computing and Information Systems. He has teaching and research interests in the use of computer algebra systems in teaching and training for research students using the WWW and CAL. He has been involved in a number of government funded projects aimed at bringing technology into the teaching of mathematics and in TEMPUS European projects with Bulgaria, Hungary and currently with Estonia and the Ukraine.

Douglas Curran obtained his BSc degree from the University of Dundee in 1969 and his PhD from the CNAA in 1983. Since 1971 he has been a Lecturer/Senior lecturer at the University of Sunderland (formerly Sunderland Polytechnic). He has teaching and research interests in the use of computer algebra systems in teaching, dynamical systems models in economics and the numerical solution of differential equations. He has been involved in a number of TEMPUS European projects with Bulgaria, Hungary and currently with Estonia and the Ukraine.