Dowling20.wxmx: Dynamic Optimization

TABLE OF CONTENTS

(%i2)
```
load(draw)$  set_draw_defaults(line_width=2, grid = [2,2], point_type = filled_circle,
          head_type = 'nofilled, head_angle = 20, head_length = 0.5,
            background_color = light_gray,  draw_realpart=false)$
```

(%i3)
```
load ("Econ2.mac");
```

(%o3) *c:/work5/Econ2.mac*

1 Preface

Dowling20.wxmx is one of a number of wxMaxima files available in the section
Economic Analysis with Maxima
on my CSULB webpage.

In Dowling20.wxmx, we use Maxima to discuss the optimization of an integral over time containing functions (of time) whose form is sought using the calculus of variations, following Dowling's Chapter 20: "The Calculus of Variations". Dowling's Sec. 8, "Applications to Economics" is mainly treated as a symbolic problem, with numerical examples. Maxima functions written for this chapter are in our file Econ2.mac.

We have changed some of the symbols used in particular problems. An approximate pdf translation (using Microsoft print to pdf) is available as Dowling20fit.pdf. That pdf file can be searched using Ctrl-F.

Ted Woollett
https://home.csulb.edu/~woollett/
woollett@charter.net
April 7, 2022

2 References

Introduction to Mathematical Economics, 3rd ed., Edward T. Dowling, 2012, Schaum's Outline Series, McGraw-Hill.

Fundamental Methods of Mathematical Economics, Alpha C. Chiang and Kevin Wainwright, 4th ed., 2005, McGraw-Hill

Elements of Dynamic Optimization, Alpha C. Chiang, 2012, Waveland Press

3 The Calculus of Variations and Euler's Equation

Quoting roughly Dowling's Sec. 20.1:
"In the *static* optimization problems studied in Chapters 4 and 5, we sought a *point* or *points* that would maximize a given function [for example, profit] at a particular point or period of time. Given a function y = y(x), the first-order condition for an optimal point x* is simply y'(x*) = 0. In *dynamic optimization* we seek a *curve* x*(t) which will maximize or minimize a given integral expression [an integral over time]."

"...In brief, assuming a time period from t0 = 0 to t1 = T and using xp to represent the first derivative of x with respect to time: xp = dx/dt, we seek to maximize or minimize the integral:
J = integrate (F[t, x(t), xp(t)], t, 0, T)
having fixed endpoint limits of integration, and assuming that F is a continuous function of t, x, and xp, F has continuous partial derivatives with respect to t, x, and xp."

"An integral such as J which assumes a numerical value for each of a class of functions x(t) is called a *functional*. A curve x*(t) that maximizes or minimizes the value of a functional is called an *extremal*. Acceptable candidates for an extremal are the class of functions x(t) which are continuously differentiable on the given time interval and which typically satisfy some fixed endpoint conditions."

3.1 Example 1

"A firm wishing to maximize profits π from time t0 = 0 to t1 = T finds that demand for its product depends on not only the price p of the product but also the rate of change of the price dp/dt. By assuming that costs are fixed and that both p and dp/dt are functions of time, and using pd to stand for dp/dt (pd = "p-dot"), the objective is to maximize the numerical value of the integral
J1 = integrate (π[t, p(t), pd(t)], t, 0, T)."

"A second firm has found that its total cost C depends on the level of production x(t) and the rate of change of production dx/dt, represented by xp (for x-prime), due to start-up and tapering-off costs. Assuming this second firm wishes to minimize costs, and that x and xp are functions of time, the objective is to minimize the numerical value of the integral
J2 = integrate (C[t, x(t), xp(t)], t, 0, T), subject to: x(t0) = x0, and x(T) = xT.
These initial and terminal constraints are known as 'endpoint conditions'."

3.2 Euler's Equation: The Necessary Condition

"For a curve x*(t) connecting the points (t0, x0) and (t1, x1) to be an 'extremal' for (i.e., to optimize) the functional (here xp stands for dx/dt):

J = integrate (F[t, x(t), xp(t)],

the necessary condition, called Euler's equation, is

∂F/∂x = d/dt (∂F/∂xp).

Using the chain rule to take the total derivative of ∂F/∂xp with respect to time t, and omitting the arguments for simplicity, we get on the right hand side:

d/dt (∂F/∂xp) = ∂ (∂F/∂xp)/∂t + ∂ (∂F/∂xp)/∂x * dx/dt + ∂ (∂F/∂xp)/∂xp * d²x/dt²,

or using subscript-like notation for partial derivatives and xpp for d²x/dt²,

d/dt (∂F/∂xp) = Fxp,t + Fxp,x*xp + Fxp,xp*xpp,

we can write Euler's equation in the explicit form:

∂F/∂x == Fx = Fxp,t + Fxp,x*xp + Fxp,xp*xpp."

Euler's equation, in most cases, produces a second order ordinary differential equation for x(t), which the candidate extremal must satisfy. In Example 3, Euler's equation leads to the candidate extremal immediately, without any further integration.

4 Example 3

Optimize the functional of x(t) represented by the definite integral:
J = integrate (6*x^2*exp(3*t) + 4*t*xp, t, 0, T).

We first work this "by hand", and then use our Extremal function.

Using the expanded form of Euler's equation, we first find the needed derivatives.
F (t, x, xp) = 6*x^2*exp(3*t) + 4*t*xp.

Fx == ∂F/∂x = 12*x*exp(3*t),
Fxp == diff (F, dx/dt) = 4*t,
Fxp,t == ∂ (Fxp)/∂t = 4,
Fxp,x == ∂ (Fxp)/∂x = 0,
Fxp,xp == ∂ (Fxp)/∂xp = 0.

Then, assembling the pieces:
12*x*exp(3*t) = 4 + 0 = 4,
whence we can solve for x(t): x = (1/3)*exp(- 3*t), which is the candidate extremal without any further integration. This satisfies the necessary condition for dynamic optimization, which only makes the solution a candidate for an extremal. The sufficiency conditions, which follow, must also be applied.

4.1 Using Extremal (F(t, x, xp))

The Maxima function Extremal (F(t, x, xp)) is defined in Econ2.mac. The code requires the use of t, x, xp as variables. Using this function for Ex. 3:

```
(%i4) Extremal (6*x^2*exp(3*t) + 4*t*xp);
```

(%o4) $x = \frac{\%e^{-3\,t}}{3}$

which agrees with our hand solution.

5 *Example 4*

Optimize the functional of x(t) represented by the definite integral:
J = integrate (4*xp^2 + 12*x*t - 5*t, t, 0, 2),
subject to x(0) = 1 and x(2) = 4.

Here, F (t, x(t), xp(t)) = 4*xp^2 + 12*x*t - 5*t.

We start with a hand calculation. Using the expanded form of Euler's equation, we first find the partial derivatives needed:
  Fx == ∂F/∂x = 12*t,
  Fxp == diff (F, dx/dt) = 8*xp,
  Fxp,t == ∂ (Fxp)/∂t = 0,
  Fxp,x == ∂ (Fxp)/∂x = 0,
  Fxp,xp == ∂ (Fxp)/∂xp = 8.
  Then, assembling the pieces:
    12*t = 8*xpp,
whence we can solve for x(t): x = t^3/4 + k1*t + k2, which is the candidate extremal.
This satisfies the necessary condition for dynamic optimization, which only makes the solution a candidate for an extremal. The sufficiency conditions, which follow, must also be applied.

When Euler's equation includes a term proportional to xpp == d²x/dt², the differential equation is displayed (in terms of xpp) by Extremal.

(%i5) Extremal (4*xp^2 + 12*x*t - 5*t);

$$ode: \quad xpp = \frac{3\,t}{2}$$

(%o5) $$x = \frac{t^3}{4} + \%k2\,t + \%k1$$

(%i6) grind(%)$

x = t^3/4+%k2*t+%k1$

6 Sufficient Conditions for an Optimum Solution x*(t)

Quoting Dowling, Sec. 20.5:
"Assuming the necessary conditions for an extremal are satisfied [x(t) satisfies Euler's Equation],
1. If the functional F(t, x(t), xp(t)) is jointly concave in x(t) and xp(t), then the necessary conditions are sufficient for a maximum.
2. If the functional F(t, x(t), xp(t)) is jointly convex in x(t) and xp(t), then the necessary conditions are sufficient for a minimum.

Joint concavity and convexity are easily determined in terms of the sign definiteness of the quadratic form of the second derivatives of the functional F. Given the discriminant matrix
D = matrix ([Fx,x, Fx,xp], [Fxp,x, Fxp,xp]),

1. a) If Fx,x < 0 and |D| > 0, D is the discriminant of a negative definite quadratic form and F is strictly concave, making the extremal a global maximum.
b) If Fx,x <= 0 and |D| >= 0, when tested for both possible ordering of the variables x, xp, D is the discriminant of a negative semidefinite quadratic form and F is simply concave, which is sufficient for a relative maximum.

2. a) If Fx,x > 0 and |D| > 0, D is the discriminant of a positive definite quadratic form and F is strictly convex, making the extremal a global mimimum.
b) If Fx,x >= 0 and |D| >= 0, when tested for both possible ordering of the variables x, xp, D is the discriminant of a positive semidefinite quadratic form and F is simply convex, which is sufficient for a relative minimum.

6.1 Using NumSuffCond (F(t, x, xp))

The Maxima function NumSuffCond (F (t, x, xp)), defined in Econ2.mac, uses the standard Maxima function hessian to define the discriminant matrices H1 : hessian (F, [x, xp]) and H2 : hessian (F, [xp, x]), to evaluate the signs of D11 : H1[1,1] and D1 : determinant (H1) and similarly for H2, and then calls NumSufficient (D11, D1, D21, D2) for analysis of the nature of the candidate extremal.

You must use the variables t, x, xp for this function. This function prints either 'global maximum', 'relative maximum', 'global minimum', 'relative minimum', or 'neither maximized nor minimized: saddle point'.

In Example 3 we have F (t, x, xp) = 6*x^2*exp(3*t) + 4*t*xp. We can first form the discriminant matrix by hand and then take the determinant:

```
(%i7) H1 : hessian(6*x^2*exp(3*t) + 4*t*xp, [x, xp]);
```

(H1) $$\begin{pmatrix} 12\,\%e^{3t} & 0 \\ 0 & 0 \end{pmatrix}$$

```
(%i8) D1 : determinant(H1);
```

(D1) 0

(%i9) H2 : hessian(6*x^2*exp(3*t) + 4*t*xp, [xp, x]);

(H2) $\begin{pmatrix} 0 & 0 \\ 0 & 12\,\%e^{3\,t} \end{pmatrix}$

(%i10) D2 : determinant(H2);

(D2) 0

In Ex. 3 we have Fx,xp = 12*exp(3*t) > 0 and D1 = 0, and looking at the other possible ordering of the variables (x, xp), we have Fxp,x = 0 and D2 = 0, which together imply a positive semidefinite case and a simply convex F, a sufficient condition for a relative minimum.

Using our Maxima function NumSuffCond:

(%i11) NumSuffCond (6*x^2*exp(3*t) + 4*t*xp)$

d11 $=12\,\%e^{3\,t}$

relative minimum

For Ex. 4 we also get sufficient conditions for a relative minimum.

(%i12) NumSuffCond(4*xp^2 + 12*x*t - 5*t)$

relative minimum

Here is an example of a case in which no sufficient conditions for an optimum are present:

(%i13) NumSuffCond (5*x^2 + 27*x - 8*x*xp - xp^2)$

neither maximized nor minimized: saddle point

6.2 Using NumDynamic (F(t, x, xp))

The syntax of NumDynamic, defined in Econ2.mac, requires the use of (t, x, xp) variables, in which x(t) is the desired solution, and xp (x-prime) means the first derivative dx/dt = x'(t). In the code output, xpp stands for x''(t), the second derivative of x(t) with respect to the time t. This function first calls Extremal, and then calls NumSuffCond.

(%i14) fundef (NumDynamic);

(%o14) NumDynamic(*F*):=block([*cdex*],*cdex*:Extremal(*F*),
print(*candidate extremal:* ,*cdex*),NumSuffCond(*F*),*done*)

Using NumDynamic(F) for Ex. 3:

(%i15) NumDynamic (6*x^2*exp(3*t) + 4*t*xp)$

candidate extremal: $x = \frac{\%e^{-3t}}{3}$

$d11 = 12\ \%e^{3t}$

relative minimum

Using NumDynamic(F) for Ex. 4:

(%i16) NumDynamic (4*xp^2 + 12*x*t - 5*t)$

ode: $xpp = \frac{3t}{2}$

candidate extremal: $x = \frac{t^3}{4} + \%k2\, t + \%k1$

relative minimum

Using NumDynamic with Dowling prob 20.15:

(%i17) NumDynamic (5*x^2 + 27*x - 8*x*xp - xp^2)$

ode: $xpp + 5x = -\frac{27}{2}$

candidate extremal: $x = \%k1 \sin(\sqrt{5}\, t) + \%k2 \cos(\sqrt{5}\, t) - \frac{27}{10}$

neither maximized nor minimized: saddle point

6.3 Using NumSuffCond (F) with details = true

The code file Econ2.mac sets details : false, which is then the "default" behavior. However, by setting details to true, you are able to see the values of the various Hessian matrices and the resulting discriminants (until you reset details to false).

(%i20) details : true$
NumSuffCond (5*x^2 + 27*x - 8*x*xp - xp^2)$
details : false$

$H1 = \begin{pmatrix} 10 & -8 \\ -8 & -2 \end{pmatrix}$

$D11 = 10$

$D1 = -84.0$

$H2 = \begin{pmatrix} -2 & -8 \\ -8 & 10 \end{pmatrix}$

$D21 = -2$

$D2 = -84.0$

neither maximized nor minimized: saddle point

6.4 Numerical Practice Problems

Dowling prob 20.14

(%i21) NumDynamic (7*xp^2 + 4*x*xp - 63*x^2)$

ode: $xpp + 9\,x = 0$

candidate extremal: $x = \%k1\,\sin(3\,t) + \%k2\,\cos(3\,t)$

neither maximized nor minimized: saddle point

Dowling prob 20.13

(%i22) NumDynamic (16*x^2 + 9*x*xp + 8*xp^2)$

ode: $xpp - 2\,x = 0$

candidate extremal: $x = \%k1\,\%e^{\sqrt{2}\,t} + \%k2\,\%e^{-\sqrt{2}\,t}$

global minimum

Dowling prob 20.12

(%i23) NumDynamic (- 16*x^2 + 144*x + 11*x*xp - 4*xp^2)$

ode: $xpp - 4\,x = -18$

candidate extremal: $x = \%k1\,\%e^{2\,t} + \%k2\,\%e^{-2\,t} + \frac{9}{2}$

global maximum

Dowling prob 20.11

(%i24) NumDynamic (15*x^2 - 132*x + 19*x*xp + 12*xp^2)$

$$ode: \quad xpp - \frac{5\,x}{4} = -\frac{11}{2}$$

$$candidate\ extremal: \quad x = \%k1\ \%e^{\frac{\sqrt{5}\,t}{2}} + \%k2\ \%e^{-\frac{\sqrt{5}\,t}{2}} + \frac{22}{5}$$

global minimum

Dowling prob 20.16

(%i25) NumDynamic (exp (0.12*t)*(5*xp^2 - 18*x))$

$$ode: \quad xpp + 0.12\ xp = -\frac{9}{5}$$

$$candidate\ extremal: \quad x = \%k2\ \%e^{-\frac{3\,t}{25}} - 15\,t + \%k1 + 125$$

$$d21 = 10\ \%e^{0.12\,t}$$

relative minimum

We can clearly redefine %k1 (arbitrary so far) to write the last two terms as %k1.

Dowling prob 20.17

(%i26) NumDynamic (exp (- 0.05*t)*(4*xp^2 + 15*x))$

$$ode: \quad xpp - 0.05\ xp = \frac{15}{8}$$

$$candidate\ extremal: \quad x = \%k1\ \%e^{t/20} - \frac{75\,t + 1500}{2} + \%k2$$

$$d21 = 8\ \%e^{-0.05\,t}$$

relative minimum

Again we can combine %k2 - 750 as %k2.

Because 8*exp(- 0.05*t) depends on the parameter t, Maxima regards the whole expression as non-numerical, and numberp(expr) returns false. Inside our Maxima function NumSufficient(d11,d1,d21,d2) such arguments are displayed to the user, as we see just above, where d21 is Fxp,xp = diff (F, xp, 2), and we may well distrust the logic of the code being used.

(%i27) numberp (8*exp(- 0.05*t));

(%o27) ***false***

We can double-check the sufficient condtions ourselves by setting details to true.

(%i30) details : true$
NumDynamic (exp (- 0.05*t)*(4*xp^2 + 15*x))$
details : false$

$ffx = 15\ \%e^{-0.05\ t}$

$ffxp = 8\ \%e^{-0.05\ t}\ xp$

$ffxpt = -0.4\ \%e^{-0.05\ t}\ xp$

$ffxpx = 0$

$ffxpxp = 8\ \%e^{-\frac{t}{20}}$

$varOde = -8\ \%e^{-\frac{t}{20}}\ xpp + 0.4\ \%e^{-0.05\ t}\ xp + 15\ \%e^{-0.05\ t}$

$varOde = xpp - 0.05\ xp - \frac{15}{8}$

$AA = -0.05$

$BB = 0$

$CCL = [\frac{15}{8}]$

$CC = \frac{15}{8}$

$ode:\quad xpp - 0.05\ xp = \frac{15}{8}$

$candidate\ extremal:\quad x = \%k1\ \%e^{t/20} - \frac{75\ t + 1500}{2} + \%k2$

$H1 = \begin{pmatrix} 0 & 0 \\ 0 & 8\ \%e^{-0.05\ t} \end{pmatrix}$

$D11 = 0$

$D1 = 0.0$

$H2 = \begin{pmatrix} 8\ \%e^{-0.05\ t} & 0 \\ 0 & 0 \end{pmatrix}$

$D21 = 8\ \%e^{-0.05\ t}$

$D2 = 0.0$

$d21 = 8\ \%e^{-0.05\ t}$

relative minimum

Since Fx,x = 0, determinant (H1) = 0, Fxp,xp > 0, and determinant (H2) = 0, we have sufficient conditions for a relative minimum.

7 Symbolic Economic Applications

7.1 Minimizing production and inventory costs over time.

Quoting Dowling in his Sec. 20.8:
"A firm wishes to minimize the present value at discount rate r of an order of N units to be delivered at time T. The firm's costs consist of production costs a*(dx/dt)^2 and inventory costs b*x(t), where a and b are positive constants; x(t) is the accumulated inventory by time t; the rate of change of inventory is the 'production rate" xp = dx/dt, where dx/dt >= 0; and a*dx/dt == a*xp is the per unit cost of production. Assuming x(0) = 0 and the firm wishes to achieve x(T) = N, in terms of the calculus of variations the firm must:
minimize J = integrate (exp(- r*t)*(a*xp^2 + b*x), t, 0, T)
subject to: x(0) = 0 and x(T) = N."

(%i31) xsoln : Extremal (exp (-r*t)*(a*xp^2 + b*x));

$$ode:\ \ xpp - r\,xp = \frac{b}{2\,a}$$

*Is r zero or nonzero?*nonzero;

(xsoln) $$x = \%k1\,\%e^{r\,t} - \frac{b\,r\,t + b}{2\,a\,r^2} + \%k2$$

(%i32) sx : rhs (xsoln);

(sx) $$\%k1\,\%e^{r\,t} - \frac{b\,r\,t + b}{2\,a\,r^2} + \%k2$$

(%i33) ksolns : solve ([at (sx, t = 0) = 0, at (sx, t = T) = N], [%k1, %k2]);

(ksolns) $$[[\%k1 = \frac{2\,N\,a\,r + T\,b}{a\,r\,(2\,\%e^{T\,r} - 2)}, \%k2 = -\frac{b\,(1 - \%e^{T\,r}) + 2\,N\,a\,r^2 + T\,b\,r}{a\,r^2\,(2\,\%e^{T\,r} - 2)}]]$$

(%i34) ksolns : ksolns[1];

(ksolns) $$[\%k1 = \frac{2\,N\,a\,r + T\,b}{a\,r\,(2\,\%e^{T\,r} - 2)}, \%k2 = -\frac{b\,(1 - \%e^{T\,r}) + 2\,N\,a\,r^2 + T\,b\,r}{a\,r^2\,(2\,\%e^{T\,r} - 2)}]$$

(%i35) sx : at (sx, ksolns);

(sx) $$\frac{(2\,N\,a\,r + T\,b)\,\%e^{r\,t}}{a\,r\,(2\,\%e^{T\,r} - 2)} - \frac{b\,r\,t + b}{2\,a\,r^2} - \frac{b\,(1 - \%e^{T\,r}) + 2\,N\,a\,r^2 + T\,b\,r}{a\,r^2\,(2\,\%e^{T\,r} - 2)}$$

(%i36) ratsimp (sx);

(%o36) $$\frac{(2\,N\,a\,r + T\,b)\,\%e^{r\,t} + (b - b\,\%e^{T\,r})\,t - 2\,N\,a\,r - T\,b}{2\,a\,r\,\%e^{T\,r} - 2\,a\,r}$$

Dowling has the solution for x(t) in the form of sxcompare:

(%i37) sxcompare : (N + b*T/(2*a*r))*(exp(r*t) - 1)/(exp(r*T) - 1) - b*t/(2*a*r);

(sxcompare) $$\frac{\left(\frac{T\,b}{2\,a\,r}+N\right)(\%e^{r\,t}-1)}{\%e^{T\,r}-1}-\frac{b\,t}{2\,a\,r}$$

Here we verify that our solution sx is algebraically the same, if both comparison expressions are expanded out and then the difference is taken.

(%i38) is (equal (sx, sxcompare));

(%o38) ***true***

(%i39) ratsimp (expand (sx - sxcompare));

(%o39) 0

7.2 Prob. 20.19: Dynamic Maximization of Firm's Profit

The demand for a monopolist's product in terms of the number of units x(t) she can sell per unit time depends on both the price p(t) of the good and the rate of change of the price dp/dt (we will use the symbol pd (for p-dot) in Maxima to stand for dp/dt:
    x(t) = a p(t) + b dp/dt + c.

Production costs per unit time z(x) as a function of the rate of production x are assumed to be given by   z(x) = m x^2 + n x + k.

We will assume a < 0, and m > 0, as one would expect from economic theory, independently of the sign of b.

(%i40) z : m*x^2 + n*x + k;

(z) $m\,x^2+n\,x+k$

The firm's revenue per unit time F is price per unit (p) times the number of units sold per unit time (x), and the problem is to determine the time path of price per unit p(t) by maximizing the integral of F(t, p, pd) over some finite specified time interval, in which F stands for the time dependent profit per unit time, given by the difference of revenue per unit time and costs per unit time. The product F*dt then yields the profit over the infinitesimal time interval dt and the finite time integral: integrate (F, t, 0, T) gives the profit over the time interval (0,T). Because this is not a purely numerical dynamical optimization problem, we work out the calculation steps "by hand", arriving at a symbolic Euler's Equation which will lead to an optimal p(t).

(%i41) F : p*x - z;

(F) $-m\,x^2+p\,x-n\,x-k$

Here we insert our assumption about x as a function of p and dp/dt.

(%i42) F : at (F, x = a*p + b*pd + c), expand;

(F) $-b^2\, m\, pd^2 - 2\, a\, b\, m\, p\, pd + b\, p\, pd - b\, n\, pd - 2\, b\, c\, m\, pd - a^2\, m\, p^2 + a\, p^2 - a\, n\, p - 2\, a\, c\, m\, p + c\, p - c\, n - c^2\, m - k$

Let Fp stand for the partial derivative of F(t, p, pd) with respect to p.

(%i43) Fp : diff (F, p);

(Fp) $-2\, a\, b\, m\, pd + b\, pd - 2\, a^2\, m\, p + 2\, a\, p - a\, n - 2\, a\, c\, m + c$

Let Fpd stand for the partial derivative of F(t, p, pd) with respect to pd.

(%i44) Fpd : diff (F, pd);

(Fpd) $-2\, b^2\, m\, pd - 2\, a\, b\, m\, p + b\, p - b\, n - 2\, b\, c\, m$

The Euler equation can be written in the form
Fp = Fpd,t + Fpd,p*pd + Fpd,pd*pdd,
where pdd stands for $\partial^2 p/\partial t^2$ (p-double-dot).

Let Fpdt stand for the partial derivative of Fpd with respect to time t, Fpd,t.

(%i45) Fpdt : diff (Fpd, t);

(Fpdt) 0

Let Fpdp stand for the partial derivative of Fpd with respect to p, Fpd,p.

(%i46) Fpdp : diff (Fpd, p);

(Fpdp) $b - 2\, a\, b\, m$

Let Fpdpd stand for the partial derivative of Fpd with respect to pd, Fpd,pd.

(%i47) Fpdpd : diff (Fpd, pd);

(Fpdpd) $-2\, b^2\, m$

If we bring all terms over to the left hand side, we get the Euler equation in the form
eqn = 0.

(%i48) ode : Fp - Fpdt - Fpdp*pd - Fpdpd*pdd, expand;

(ode) $2\, b^2\, m\, pdd - 2\, a^2\, m\, p + 2\, a\, p - a\, n - 2\, a\, c\, m + c$

We want to identify the quantities A, B, C if the second order ode is written in the form
d²p/dt² + A dp/dt + B p = C, or
pdd + A*pd + B*p = C.

First, divide all terms by the coefficient of pdd

```
(%i49) ode : expand (ode/coeff (ode,pdd));
```

(ode) $$pdd+\frac{a\,p}{b^2\,m}-\frac{a^2\,p}{b^2}-\frac{a\,n}{2\,b^2\,m}+\frac{c}{2\,b^2\,m}-\frac{a\,c}{b^2}$$

There is no dp/dt term, so A = 0. We next pick off B as the coeffecient of p.

```
(%i50) B : ratsimp (coeff (ode, p));
```

(B) $$-\frac{a^2\,m-a}{b^2\,m}$$

Factoring out a in the numerator of B, we get B = - a*(m*a -1)/(b^2*m).

Because we assume a < 0 and m > 0, m*a < 0, (m*a - 1) < 0, and a*(m*a - 1) > 0, so B < 0.

We can use part (expr, j) to get access to individual parts in a sum of terms.

```
(%i51) part (ode, 4);
```

(%o51) $$-\frac{a\,n}{2\,b^2\,m}$$

We can then use sum (expr, j, jstart, jend) to get the remaining terms (with a minus sign) which define C:

```
(%i52) C : - sum (part (ode,j), j, 4, 6);
```

(C) $$\frac{a\,n}{2\,b^2\,m}-\frac{c}{2\,b^2\,m}+\frac{a\,c}{b^2}$$

With our second order differential equation for p(t) reduced to the form
$d^2p/dt^2 + B*p = C$,
we can use our basic rules, p is the sum of the solution (pc) of the complementary equation
$d^2pc/dt^2 + B*pc = 0$
and the particular solution (pp), which in this case is
pp = C/B.
Returning to finding pc, we use the trial solution pc = exp(r*t) to get the characteristic equation r^2 + B = 0, or r = +/- sqrt(- B). Recall that we found B < 0 above, so - B > 0 and we have a double real root, +/- R, where R = sqrt (a*(m*a -1)/(b^2*m)).

The complementary solution is then
pc(t) = k1*exp(R*t) + k2*exp (-R*t).

(%i53) pp : C/B, ratsimp;

(pp) $$-\frac{a\,n+2\,a\,c\,m-c}{2\,a^2\,m-2\,a}$$

Absorbing the overall minus sign into the denominator, and factoring the denominator, the particular solution has the form
pp = (2*m*a*c + n*a - c)/(2*a)*(1 - m*a).

Then the predicted profit as a function of time is

p(t) = k1*exp(R*t) + k2*exp (-R*t) + (2*m*a*c + n*a - c)/(2*a)*(1 - m*a),

which agrees with Dowling's solution. In our solution, k1 and k2 are arbitrary constants, which can be assigned numerical values given p(0) and pd(0), for example.

7.3 Prob. 20.20: Maximize Stream of Utility U(C(t))

With K(t) the capital available for production at time t, and with G(K(t)) the value of the resulting production at time t (G(K) is an unspecified function), and with C(t) the amount of consumption at time t, and dK/dt the investment made at time t, we start with the assumption that the fruits of production are divided between consumption and investment (in capital for increased production):
G(K(t) = C(t) + dK/dt
so that
C(t) = G(K(t) - dK/dt, which implies ∂C/dK = dG/dK.

We also assume some unspecified function U(C) is adopted which reflects the desireability of a given level of consumption (we call U(C) the instantaneous utility from the flow of consumption C(t)), and we seek a differential equation for K(t) which will maximize the integral
I = ∫ U(C(t)) dt
taken over some finite time interval [t0, tf], such that the capital K available for production takes on specified end point values K(t0) = K0 and K(tf) = Kf.

Letting Kd stand for dK/dt and Kdd stand for d^2K/dt^2, and letting U' stand for dU/dC and G' stand for dG/dK, the variational Euler equation is,

with F(t,K,Kd) = U(C(t)) = U(G(K) - dK/dt) = U(G(K) - Kd),
∂F/∂K = ∂ (∂F/∂Kd)/∂t + ∂ (∂F/∂Kd)/∂K * Kd + ∂ (∂F/∂Kd)/∂Kd * Kdd.

To make use of Euler's equation we need the derivatives:

F_K = ∂F/∂K = dU/dC * ∂C/∂K = U' * dG/dK = U' * G',
F_Kd = ∂F/∂Kd = dU/dC * ∂C/∂Kd = dU/dC * (-1) = - U',
∂ (∂F/∂Kd)/∂t = 0,
F_{Kd, K} = ∂ (∂F/∂Kd)/∂K = d (F_Kd)/dC * ∂C/∂K = - U'' * G'
F_{Kd, Kd} = ∂ (∂F/∂Kd)/∂Kd = d (F_Kd))/dC * ∂C/∂Kd = - d(- U')/dC = U''.

Euler's equation for the candidate extremal K(t) then becomes
U'(C)*G'(K) = - U''(C)* (G'(K)*dK/dt - d^2K/dt^2).
The result, given the functions U(C) and G(K), would then be a second order ordinary differential equation for K(t) which maximizes the given integral I.

7.4 Prob. 20.21: Maximize Discounted Stream of Utility U(C(t))

We alter the expression assumed for production capital investment from dK/dt to investment = dK/dt + b*K(t), where b is a constant "rate of capital depreciation", so that now the consumption C = G(K) - dK/dt - b*K(t), so ∂C/∂K = G'(K) - b, and ∂C/∂Kd = -1.

We seek a differential equation for K(t) whose solution will maximize the discounted stream of utility U(C) from consumption over the finite time interval 0 <= t <= T:
I = ∫ exp(-r*t)*U(C(t)) dt = ∫ F(t, K, dK/dt) dt.

F_K = ∂F/∂K = exp(-r*t)*dU/dC*∂C/∂K = exp(-r*t)*U'(C)*(G'(K) - b),
F_Kd = ∂F/∂Kd = exp(-r*t)*dU/dC * ∂C/∂Kd = exp(-r*t)*dU/dC * (-1) = - exp(-r*t)*U'(C),
F_{Kd, t} = ∂ (∂F/∂Kd)/∂t = r*exp(-r*t)*U'(C),
F_{Kd, K} = ∂ (∂F/∂Kd)/∂K = - exp(-r*t)*U''(C)*(G'(K) - b),
F_{Kd, Kd} = ∂ (∂F/∂Kd)/∂Kd = exp(-r*t)*U''(C).

Substituting into Euler's equation:
exp(-r*t)*U'(C)*(G'(K) - b) = r*exp(-r*t)*U'(C) - exp(-r*t)*U''(C)*(G'(K) - b)*Kd +
exp(-r*t)*U''(C)*Kdd.

Cancelling the common factor exp(-r*t), we get
U'(C)*(G'(K) - b) = r*U'(C) - U''(C)*(G'(K) - b)*Kd + U''(C)*Kdd, or

U'(C) * (G'(K) - b) = r*U'(C) - U''(C) * (G'(K)*Kd - b*Kd - Kdd).

Without specification of the functions U(C) and G(K), we cannot proceed further. However if we write the variational Euler's equation in the alternative (and equivalent) form involving the total derivative with respect to the time t:

∂F/∂K = d (∂F/∂Kd) /dt

we can write:

exp(-r*t)*U'(C)*(G'(K) - b) = - d (exp(-r*t)*U'(C))/ dt
= r*exp(-r*t)*U'(C) - exp(-r*t)* d (U'(C))/dt.

Cancelling the common factor exp (-r*t), we get

U'(C)*(G' - b) = r*U'(C) - d (U'(C))/dt, or
d (U'(C))/dt = U'(C)*(r + b - G'), or
(1/U')*d U' /dt = (r + b - G'), or
d (ln (U'))/dt = (r + b - G'),

where the term on the left, the rate of change of the natural logarithm of the marginal utility U'(C), equals the discount rate plus the depreciation rate minus the marginal product of capital G'(K).

In brief, if we consider the term on the left as capital gains, the optimal time path of capital K(t) suggests that if capital gains are greater than the discount rate plus the depreciation rate minus the marginal product of capital, then more capital and hence more consumption will be forthcoming. If it is less, then capital accumulation and consumption will be scaled back.

Note that Maxima's log(y) means the natural logarithm of y.
Maxima knows that diff (log(y), t) = (1/y)*dy/dt:

```
(%i54) log(exp(a));
```

(%o54) a

```
(%i56) depends(y, t);
       diff (log (y), t);
```

(%o55) $[y(t)]$

(%o56) $$\frac{\frac{d}{d\,t}\,y}{y}$$

```
(%i57) remove (y, dependency);
```

(%o57) $done$

7.5 Prob. 20.22: Specific Model for Prob. 20.21

Extending Prob. 20.21, we assume U(C) = C^n, where 0 <= n <= 1.
Again we have C(t) = G(K) - I(t), with I(t) = production capital investment.
We also assume G(K) = a*K(t) with a > 0.
Finally we assume capital investment for further production takes the form:
I(t) = dK/dt + B + b*K(t), with 0 <= b =< 1 and B > 0.
This last assumption comes from comes from the statement:
(increase in K stock) = investment - (linear depreciation), or
dK/dt = I(t) - (B + b*K(t)).

We now have F = exp(-r*t)*U(C) = exp(-r*t)*C^n and
C = a*K - dK/dt - B - b*K(t),
so F (t, K, Kd) = exp (-r*t)*(a*K - Kd - B - b*K)^n, or defining
m = a - b,
F (t, K, Kd) = exp (-r*t)*(m*K - Kd - B)^n

Then the needed derivatives are
F_K = n*m*exp (-r*t)*(m*K - Kd - B)^(n - 1),
F_Kd = - n*exp (-r*t)*(m*K - Kd - B)^(n - 1),
F_{Kd, t} = n*r*exp (-r*t)*(m*K - Kd - B)^(n - 1),
F_{Kd, K} = - m*n*(n - 1)*exp (-r*t)*(m*K - Kd - B)^(n - 2),
F_{Kd, Kd} = n*(n - 1)*exp (-r*t)*(m*K - Kd - B)^(n - 2).

Using these derivatives in the expanded form of Euler's equation:
F_K = F_{Kd, t} + F_{Kd, K}*Kd + F_{Kd, Kd}*Kdd,
n*m*exp (-r*t)*(m*K - Kd - B)^(n - 1) = n*r*exp (-r*t)*(m*K - Kd - B)^(n - 1)
- m*n*(n - 1)*exp (-r*t)*(m*K - Kd - B)^(n - 2)* Kd
+ n*(n - 1)*exp (-r*t)*(m*K - Kd - B)^(n - 2)*Kdd.

Using x^(n-1) = x*x^(n-2), we can cancel the common factor n*exp(-r*t)*x^(n-2) to get:
m*(m*K - Kd - B) = r*(m*K - Kd - B) - m*(n - 1)*Kd + (n -1)*Kdd
= r*(m*K - Kd - B) + m*(1- n)*Kd - (1- n)*Kdd
= r*(m*K - Kd - B) + (1- n)* (m*Kd - Kdd), or
(m - r)*(m*K - Kd - B) = (1- n)* (m*Kd - Kdd).
Dividing by (1-n), we can write this as

Kdd + Z1*Kd + Z2*K = Z3,

where

Z1 = (r + m*n - 2*m)/(1 - n),
Z2 = (m^2 - r*m)/(1 - n),
Z3 = (m - r)*B/(1 - n).
Here m = a - b, a is the marginal product of capital (dG/dK = a) and
b is the constant rate of depreciation.

The exptremal solution K(t) is then Kp + Kc.

The particular solution Kp = Z3/Z2 = B/m.

The complementary solution Kc is the solution of
  d²Kc/dt² + Z1*dKc/dt + Z2*Kc = 0.

Taking as a trial solution Kc(t) = exp(R*t), R must satisfy the characteristic equation
  R^2 + Z1*R + Z2 = 0, with solutions
      R1, R2 = (-Z1 +/- sqrt (Z1^2 - 4*Z2))/2,
and Kc(t) = A1*exp(R1*t) + A2*exp(R2*t)
in terms of constants A1 and A2 which can be specified using K(t=0) and K(t=T) boundary condition values.

7.6 Prob. 20.23: Numerical Example of Prob. 20.22

Require the endpoint values K(0) = 320 and K(5) = 480.
Assume the discount rate r = 0.12, and n = 0.5, a = 0.25, B = 60, and b = 0.05.
Then m = a - b = 0.25 - 0.05 = 0.2.

7.6.1 Method 1: Substitution of numerical values

We substitute numerical values into the expressions Z1, Z2, Z3, in terms of which (as we saw just above) Euler's equation can be written: Kdd + Z1*Kd + Z2*K = Z3.

```
(%i59) kill (r, n, B, m)$
       case : [r = 0.12, n = 0.5, B = 60, m = 0.2];
(case) [r=0.12,n=0.5,B=60,m=0.2]

(%i60) Z1 : at ((r + m*n - 2*m)/(1 - n), case);
(Z1)   -0.36

(%i61) Z2 : at ((m^2 - r*m)/(1 - n), case);
(Z2)   0.032

(%i62) Z3 : at ((m - r)*B/(1 - n), case);
(Z3)   9.6

(%i63) Kp : Z3/Z2;
(Kp)   300.0
```

croots (A, B), defined in Econ2.mac, returns the pair of roots of the characteristic equation r^2 + A*r + B = 0.

(%i64) [R1, R2] : croots (Z1, Z2);

(%o64) $[\frac{1}{5},\frac{4}{25}]$

(%i65) float (%);

(%o65) $[0.2,0.16]$

(%i66) Kindef : %k1*exp(R1*t) + %k2*exp(R2*t) + Kp;

(Kindef) $\%k1\ \%e^{t/5}+\%k2\ \%e^{\frac{4\,t}{25}}+300.0$

Apply boundary conditions to determine the constants %k1 and %k2.

(%i68) eqn1 : at (Kindef, t = 0) = 320;
eqn2 : at (Kindef, t = 5) = 480;

(eqn1) $\%k2+\%k1+300.0=320$

(eqn2) $\%e^{4/5}\ \%k2+\%e\ \%k1+300.0=480$

(%i69) solns : solve ([eqn1, eqn2]), numer;

(solns) $[[\%k2=-254.97,\%k1=274.97]]$

(%i70) Kdef : at (Kindef, solns[1]);

(Kdef) $274.97\ \%e^{t/5}-254.97\ \%e^{\frac{4\,t}{25}}+300.0$

So K(t) = 274.97*exp(t/5) - 254.97*exp(4*t/25) + 300, which agrees with Dowling's answer.

7.6.2 Method 2: Starting with F(t, K, Kd)

We start with: F (t, K, Kd) = exp (-r*t)*(m*K - Kd - B)^n
= exp (- 0.12*t)*(0.2*K - Kd - 60)^0.5,
and take the needed partial derivatives to obtain Euler's equation in expanded form.

(%i71) F : exp (- 0.12*t)*(0.2*K - Kd - 60)^0.5;

(F) $(-Kd+0.2\ K-60)^{0.5}\ \%e^{-0.12\ t}$

(%i72) FK : diff (F, K);

(FK) $$\frac{0.1\ \%e^{-0.12\ t}}{(-Kd+0.2\ K-60)^{0.5}}$$

(%i73) FKd : diff (F, Kd);

(FKd) $$-\frac{0.5\ \%e^{-0.12\ t}}{(-Kd+0.2\ K-60)^{0.5}}$$

(%i74) FKdK : diff (FKd, K);

(FKdK) $$\frac{0.05\ \%e^{-0.12\ t}}{(-Kd+0.2\ K-60)^{1.5}}$$

(%i75) FKdKd : diff (FKd, Kd);

(FKdKd) $$-\frac{0.25\ \%e^{-0.12\ t}}{(-Kd+0.2\ K-60)^{1.5}}$$

(%i76) FKdt : diff (FKd, t);

(FKdt) $$\frac{0.06\ \%e^{-0.12\ t}}{(-Kd+0.2\ K-60)^{0.5}}$$

Bringing all terms to the left hand side:

(%i77) ode : FK - FKdt - FKdK*Kd - FKdKd*Kdd;

(ode) $$\frac{0.25\ Kdd\ \%e^{-0.12\ t}}{(-Kd+0.2\ K-60)^{1.5}}-\frac{0.05\ Kd\ \%e^{-0.12\ t}}{(-Kd+0.2\ K-60)^{1.5}}+\frac{0.04\ \%e^{-0.12\ t}}{(-Kd+0.2\ K-60)^{0.5}}$$

(%i78) ode : expand (ode/coeff (ode,Kdd));

(ode) $1.0\ Kdd-0.36\ Kd+0.032\ K-9.6$

(%i79) A : coeff(ode, Kd);

(A) -0.36

(%i80) B : coeff (ode,K);

(B) 0.032

(%i81) C : 9.6;

(C) 9.6

(%i82) eqn : 'diff (K, t, 2) + A*'diff (K, t) + B*K = C;

(eqn) $$\frac{d^2}{d\,t^2}K-0.36\left(\frac{d}{d\,t}K\right)+0.032\,K=9.6$$

(%i83) soln : ode2(eqn, K, t);

(soln) $$K=\%k1\,\%e^{t/5}+\%k2\,\%e^{\frac{4\,t}{25}}+300$$

(%i84) Kindef : rhs (soln);

(Kindef) $$\%k1\,\%e^{t/5}+\%k2\,\%e^{\frac{4\,t}{25}}+300$$

Solve for the constants using the values of K(t) at t = 0 and t = 5.

(%i86) eqn1 : at (Kindef, t = 0) = 320;
eqn2 : at (Kindef, t = 5) = 480;

(eqn1) $$\%k2+\%k1+300=320$$

(eqn2) $$\%e^{4/5}\,\%k2+\%e\,\%k1+300=480$$

(%i87) solns : solve ([eqn1, eqn2]),numer;

(solns) $$[[\%k2=-254.97,\%k1=274.97]]$$

(%i88) Kdef : at (Kindef, solns[1]);

(Kdef) $$274.97\,\%e^{t/5}-254.97\,\%e^{\frac{4\,t}{25}}+300$$

which agrees with our previous result.

8 Constrained Dynamic Optimization

Dowling introduces constrained dynamic optimization in Sec. 20.6 "Dynamic Optimization Subject to Functional Constraints".

The only type of functional constraint Dowling considers in an integral constraint, which is what we have in Prob. 20.25. (Other types of constraints are discussed in the text:
Alpha C. Chiang, Elements of Dynamic Optimization, Waveland Press, Ch. 6.)

8.1 Prob. 20.25

Minimize J = integrate (exp (-r*t) * (a*xp^2 + b*x), t, 0, T)
subject to: x(0) = 0, x(T) = N, KK = integrate (xp, t, 0, T) = N.

Let H (t, x, xp) = exp (-r*t) * (a*xp^2 + b*x) + λ*xp, using a Lagrange multiplier constant λ, and minimize the integral: integrate (H(t,x,xp), t, 0, T) for an arbitrary value of λ and subject to x(0) = 0, x(T) = N. After finding a solution x*(t) we use xp = dx*/dt to require that the integral: integrate (xp, t, 0, T) = N in order to fix the value of the Lagrange multiplier λ (in principle, and assuming the solution contains the parameter λ).

We require x(t) satisfy Euler's Equation as a necessary condition. In the following, ode represents Euler's Equation with all the terms brought over to the left hand side, so ode = 0.

8.1.1 Using Extremal (H)

(%i91)
```
kill(r,a,b,N,T)$
H : exp (-r*t)*(a*xp^2 + b*x) + λ*xp;
soln : Extremal (H);
```

(H) $xp\,\lambda + \%e^{-r\,t}\,(a\,xp^2 + b\,x)$

$$ode:\quad xpp - r\,xp = \frac{b}{2\,a}$$

Is r zero or nonzero? nonzero;

(soln) $x = \%k1\,\%e^{r\,t} - \frac{b\,r\,t + b}{2\,a\,r^2} + \%k2$

(%i92)
```
sx : expand (rhs (soln));
```

(sx) $\%k1\,\%e^{r\,t} - \frac{b\,t}{2\,a\,r} - \frac{b}{2\,a\,r^2} + \%k2$

Determine the constants %k1 and %k2 using x(0) = 0 and x(T) = N.

(%i93)
```
ksolns : solve ([ at (sx, t = 0) = 0,  at (sx, t = T) = N],[%k1, %k2]);
```

(ksolns) $[[\%k1 = \frac{2\,N\,a\,r + T\,b}{a\,r\,(2\,\%e^{T\,r} - 2)}, \%k2 = -\frac{b\,(1 - \%e^{T\,r}) + 2\,N\,a\,r^2 + T\,b\,r}{a\,r^2\,(2\,\%e^{T\,r} - 2)}]]$

(%i94)
```
ksolns : ksolns[1];
```

(ksolns) $[\%k1 = \frac{2\,N\,a\,r + T\,b}{a\,r\,(2\,\%e^{T\,r} - 2)}, \%k2 = -\frac{b\,(1 - \%e^{T\,r}) + 2\,N\,a\,r^2 + T\,b\,r}{a\,r^2\,(2\,\%e^{T\,r} - 2)}]$

(%i95)
```
sx : at (sx, ksolns), expand;
```

(sx)
$$\frac{2\,N\,a\,r\,\%e^{r\,t}}{2\,a\,r\,\%e^{T\,r} - 2\,a\,r} + \frac{T\,b\,\%e^{r\,t}}{2\,a\,r\,\%e^{T\,r} - 2\,a\,r} - \frac{b\,t}{2\,a\,r} + \frac{b\,\%e^{T\,r}}{2\,a\,r^2\,\%e^{T\,r} - 2\,a\,r^2} - \frac{2\,N\,a\,r^2}{2\,a\,r^2\,\%e^{T\,r} - 2\,a\,r^2} - \frac{T\,b\,r}{2\,a\,r^2\,\%e^{T\,r} - 2\,a\,r^2} - \frac{b}{2\,a\,r^2\,\%e^{T\,r} - 2\,a\,r^2} - \frac{b}{2\,a\,r^2}$$

x*(t) looks a little simpler if we put everything over a common denominator using ratsimp.

(%i96) ratsimp(sx);

(%o96) $$\frac{(2\,N\,a\,r+T\,b)\,\%e^{r\,t}+(b-b\,\%e^{T\,r})\,t-2\,N\,a\,r-T\,b}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}$$

We want to check that the integral constraint integrate (xp, t, 0, T) = N has been achieved with this solution, so let sxp = dx*/dt.

(%i97) sxp : diff(sx, t);

(sxp) $$\frac{2\,N\,a\,r^2\,\%e^{r\,t}}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}+\frac{T\,b\,r\,\%e^{r\,t}}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}-\frac{b}{2\,a\,r}$$

KK is our constraint integral relation which should equal N.

(%i98) KK : integrate(sxp, t, 0, T);

(KK) $$\frac{2\,N\,a\,r\,\%e^{T\,r}+T\,b}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}-\frac{2\,N\,a\,r+T\,b}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}$$

(%i99) ratsimp(KK);

(%o99) N

which says we have a solution which satisfies all the constraints, and the value of the Lagrange multiplier plays no role.

8.1.2 Using Lode2 (x, t, type, A, B, C)

```
(%i107) kill(r,a,b,N,T)$
        H : exp (-r*t)*(a*xp^2 + b*x) + λ*xp;
        Hx : diff (H,x);
        Hxp : diff (H, xp);
        Hxpt : diff (Hxp, t);
        Hxpx : diff (Hxp, x);
        Hxpxp : diff (Hxp, xp);
        ode : Hx - Hxpt - Hxpx*xp - Hxpxp*xpp;
```

(H) $xp\,\lambda + \%e^{-r\,t}\,(a\,xp^2 + b\,x)$

(Hx) $b\,\%e^{-r\,t}$

(Hxp) $\lambda + 2\,a\,\%e^{-r\,t}\,xp$

(Hxpt) $-2\,a\,r\,\%e^{-r\,t}\,xp$

(Hxpx) 0

(Hxpxp) $2\,a\,\%e^{-r\,t}$

(ode) $-2\,a\,\%e^{-r\,t}\,xpp + 2\,a\,r\,\%e^{-r\,t}\,xp + b\,\%e^{-r\,t}$

```
(%i108) Hxp;
```

(%o108) $\lambda + 2\,a\,\%e^{-r\,t}\,xp$

```
(%i109) Hxpxp;
```

(%o109) $2\,a\,\%e^{-r\,t}$

```
(%i110) diff (H, xp, 2);
```

(%o110) $2\,a\,\%e^{-r\,t}$

```
(%i111) ode : exp(r*t)*ode, expand;
```

(ode) $-2\,a\,xpp + 2\,a\,r\,xp + b$

Write the ode is standard form
x'' + A*x' + B*x = C,

```
(%i112) ode : ode/(-2*a), expand;
```

(ode) $xpp - r\,xp - \frac{b}{2\,a}$

So A = -r, B = 0, and C = b/(2*a), and we have A^2 > 4*B = 0, so "type" is real and we use the Maxima function Lode2 (x, t, type, A, B, C) defined in Econ2.mac, which can be used with linear 2nd order odes. Notice that λ does not appear as a parameter in our ode.

(%i113) soln : Lode2(x, t, real, -r, 0, b/(2*a));

Is r zero or nonzero? nonzero;

(soln) $x=\%k1\,\%e^{r\,t}-\frac{b\,r\,t+b}{2\,a\,r^2}+\%k2$

(%i114) sx : expand (rhs (soln));

(sx) $\%k1\,\%e^{r\,t}-\frac{b\,t}{2\,a\,r}-\frac{b}{2\,a\,r^2}+\%k2$

Determine the constants %k1 and %k2 using x(0) = 0 and x(T) = N.

(%i115) ksolns : solve ([at (sx, t = 0) = 0, at (sx, t = T) = N],[%k1, %k2]);

(ksolns) $[[\%k1=\frac{2\,N\,a\,r+T\,b}{a\,r\,(2\,\%e^{T\,r}-2)},\%k2=-\frac{b\,(1-\%e^{T\,r})+2\,N\,a\,r^2+T\,b\,r}{a\,r^2\,(2\,\%e^{T\,r}-2)}]]$

(%i116) ksolns : ksolns[1];

(ksolns) $[\%k1=\frac{2\,N\,a\,r+T\,b}{a\,r\,(2\,\%e^{T\,r}-2)},\%k2=-\frac{b\,(1-\%e^{T\,r})+2\,N\,a\,r^2+T\,b\,r}{a\,r^2\,(2\,\%e^{T\,r}-2)}]$

(%i117) sx : at (sx, ksolns), expand;

(sx) $\frac{2\,N\,a\,r\,\%e^{r\,t}}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}+\frac{T\,b\,\%e^{r\,t}}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}-\frac{b\,t}{2\,a\,r}+\frac{b\,\%e^{T\,r}}{2\,a\,r^2\,\%e^{T\,r}-2\,a\,r^2}-\frac{2\,N\,a\,r^2}{2\,a\,r^2\,\%e^{T\,r}-2\,a\,r^2}-\frac{T\,b\,r}{2\,a\,r^2\,\%e^{T\,r}-2\,a\,r^2}-\frac{b}{2\,a\,r^2\,\%e^{T\,r}-2\,a\,r^2}-\frac{b}{2\,a\,r^2}$

x*(t) looks a little simpler if we put everything over a common denominator using ratsimp.

(%i118) ratsimp(sx);

(%o118) $\frac{(2\,N\,a\,r+T\,b)\,\%e^{r\,t}+(b-b\,\%e^{T\,r})\,t-2\,N\,a\,r-T\,b}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}$

We want to check that the integral constraint integrate (xp, t, 0, T) = N has been achieved with this solution, so let sxp = dx*/dt.

(%i119) sxp : diff(sx, t);

(sxp) $\frac{2\,N\,a\,r^2\,\%e^{r\,t}}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}+\frac{T\,b\,r\,\%e^{r\,t}}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}-\frac{b}{2\,a\,r}$

KK is our constraint integral relation which should equal N.

(%i120) KK : integrate(sxp, t, 0, T);

(KK) $$\frac{2\,N\,a\,r\,\%e^{T\,r}+T\,b}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}-\frac{2\,N\,a\,r+T\,b}{2\,a\,r\,\%e^{T\,r}-2\,a\,r}$$

(%i121) ratsimp(KK);

(%o121) N

which says we have a solution which satisfies all the constraints, and the value of the Lagrange multiplier plays no role.

8.1.3 Check Sufficient Conditions

(%i122) Hxx : diff (H, x, 2);

(Hxx) 0

(%i123) Hxpxp;

(%o123) $2\,a\,\%e^{-r\,t}$

Thus Hxpxp > 0 if a > 0.

(%i124) H1 : hessian (H, [x, xp]);

(H1) $$\begin{pmatrix} 0 & 0 \\ 0 & 2\,a\,\%e^{-r\,t} \end{pmatrix}$$

(%i125) H2 : hessian (H,[xp, x]);

(H2) $$\begin{pmatrix} 2\,a\,\%e^{-r\,t} & 0 \\ 0 & 0 \end{pmatrix}$$

Since determinant(H1) = 0, determinant(H2) = 0, Hxx = 0, and Hxpxp > 0 if a > 0, we conclude we have sufficient conditions for a relative minimum if a > 0. (See our section above on "Sufficient Conditions for an Optimum Solution."

(%i126) H;

(%o126) $xp\,\lambda+\%e^{-r\,t}\,(a\,xp^2+b\,x)$

If some elements of the discriminant are symbolic (not purely numeric), then no automatic judgement about signs can be made when we call NumSuffCond (H). The 'Num' prefix emphasizes that the function works with purely numerical expressions depending on (t, x, xp).

(%i127) NumSuffCond (H);

$$d21 = 2\, a\, \%e^{-r\, t}$$

(%o127) *done*