MODULE 5.2

Euler's Method

Download

The text's website has available for download for various system dynamics tools the file *unconstrainedError* file, which contains a model for the "Error" section below.

Introduction

With system dynamics tools, we often have the choice of simulation techniques, such as Euler's Method, Runge-Kutta 2, Runge-Kutta 4, and others. These numerical methods are for solving ordinary differential equations and estimating definite integrals for which the indefinite integral does not exist. In this module, we discuss the most straightforward of these, Euler's Method.

Reasoning behind Euler's Method

In Module 3.2, "Unconstrained Growth and Decay," to simulate $dP/dt = 0.10P$ with $P_0 = 100$, we employ the following underlying equations with *INIT* meaning "initial" and *dt* representing a small change in time, Δt:

```
growth_rate = 0.10
INIT population = 100
growth = growth_rate * population
population = population + growth * dt
```

These equations, which we enter explicitly or implicitly with a model diagram, represent the following finite difference equations using **Euler's Method**:

```
growth_rate = 0.10
population(0) = 100
growth(t) = growth_rate * population(t - Δt)
population(t) = population(t - Δt) + growth(t) * Δt
```

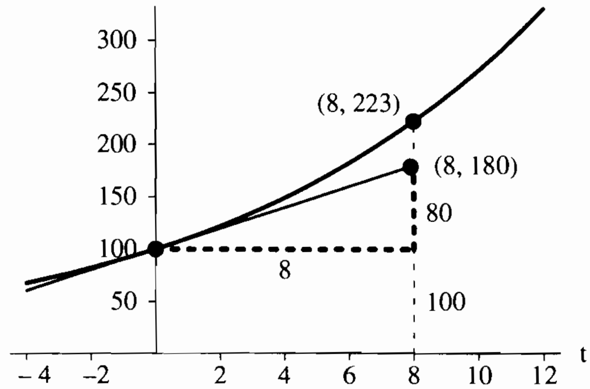

Figure 5.2.1 Actual point, (8, 223), and point obtained by Euler's Method, (8, 180)

The population at one time step, *population*(t), is the population at the previous time step, *population*$(t - \Delta t)$, plus the estimated change in population, *growth*$(t) * \Delta t$. The derivative of population with respect to time is *growth*, and

$$growth(t) = growth\_rate * population(t - \Delta t)$$
$$= 0.10 * population(t - \Delta t)$$

or $dP/dt = 0.10P$. The change in the population is the flow (in this case, *growth*) times the change in time, Δt; and the flow (*growth*) is the derivative of the function at the previous time step, $t - \Delta t$.

Figure 5.2.1 illustrates Euler's Method to estimate $P_1 = P(8)$ for the above differential equation by starting with $P_0 = P(0) = 100$ and using $\Delta t = 8$. In this situation, $t = 8$, $t - \Delta t = 0$, and the derivative at that time is $P'(0) = 0.1(100) = 10$. We multiply Δt, 8, by this derivative at the previous time step,10, to obtain the estimated change in P, 80. Consequently, the estimate for P_1 is as follows:

$$\text{Estimate for } P_1 = \text{previous value of } P + \text{estimated change in } P$$
$$= P_0 + P'(0)\Delta t$$
$$= 100 + 10(8)$$
$$= 180$$

In Module 3.2, "Unconstrained Growth and Decay," we discovered analytically that the solution to the above differential equation is $P = 100\ e^{0.1t}$. Because the graph of the actual function is concave up, this estimated value, 180, is lower than the actual value at $t = 8$, $100e^{0.1(8)} \approx 223$.

Quick Review Question 1

For $dP/dt = 10 + P/5$ with $P_0 = 500$ and $\Delta t = 0.1$, calculate the following:

a. dP/dt at $t = 0$

b. Euler's estimate of P_1

Algorithm for Euler's Method

Following the description above, Algorithm 1 presents Euler's Method.

Algorithm 1 Euler's Method

> $t \leftarrow t_0$
> $P(t_0) \leftarrow P_0$
>
> **Initialize *SimulationLength***
> **while $t <$ *SimulationLength* do the following:**
>
> $t \leftarrow t + \Delta t$
> $P(t) = P(t - \Delta t) + P'(t - \Delta t)\Delta t$

Thus, simulation uses a sequence of times—t_0, t_1, t_2, . . . —and calculates a corresponding sequence of estimated populations—P_0, P_1, P_2, ... In Algorithm 1, $t_n = t_{n-1} + \Delta t$ or $t_{n-1} = t_n - \Delta t$, and $P_n = P_{n-1} + P'(t_{n-1})\Delta t$.

However, as illustrated in Module 2.2 on "Errors," repeatedly accumulating Δt into t usually produces an accumulation error. To minimize error, we calculate the time as the sum of the initial time and an integer multiple of Δt. Using the functional notation $f(t_{n-1}, P_{n-1})$ to indicate the derivative dP/dt at step $n - 1$, Algorithm 2 presents a revised Euler's Method that minimizes accumulation of error.

Algorithm 2 Revised Euler's Method to minimize error accumulation of time with $f(t_{n-1}, P_{n-1})$ indicating the derivative dP/dt at step $n - 1$

> **Initialize t_0 and P_0**
>
> **Initialize *NumberOfSteps***
> **for n going from 1 to *NumberOfSteps* do the following:**
>
> $t_n = t_0 + n\,\Delta t$
> $P_n = P_{n-1} + f(t_{n-1}, P_{n-1})\,\Delta t$

Quick Review Question 2

Match each of the symbols below to its the meaning in Algorithm 2 for Euler's Method. Here, "previous" means "immediately previous."

A. Change in time between time steps
B. Derivative of function at estimated value of function for current time step
C. Derivative of function at estimated point for previous time step

D. Estimated value of function at current time step
E. Estimated value of function at previous time step
F. Initial time
G. Initial value of function
H. Number of current time step
I. Time at current time step
J. Time at previous time step
a. t_n
b. t_0
c. n
d. Δt
e. P_n
f. P_{n-1}
g. $f(t_{n-1}, P_{n-1})$
h. t_{n-1}

Error

As we saw in Module 3.2, "Unconstrained Growth and Decay" the analytical solution to $dP/dt = 0.10P$ with $P_0 = 100$ is $P = 100e^{0.10t}$. Even with Algorithm 2, comparison of the results of Euler's Method with the analytical solution reveals an accumulation error. As Figure 5.2.2 and an *unconstrainedError* file illustrate (see "Download"), we can adjust the unconstrained growth model to demonstrate the variation.

The converter/variable *actual_population* evaluates P_0e^{rt} or $100e^{0.10t}$. The formula does not use the changing value of *population*, but the initial population, *initial_population*. With *Time* as the current value of time and *EXP* as the exponential function, the equation in the converter *actual_population* is as follows:

```
initial_population * EXP(growth_rate * Time)
```

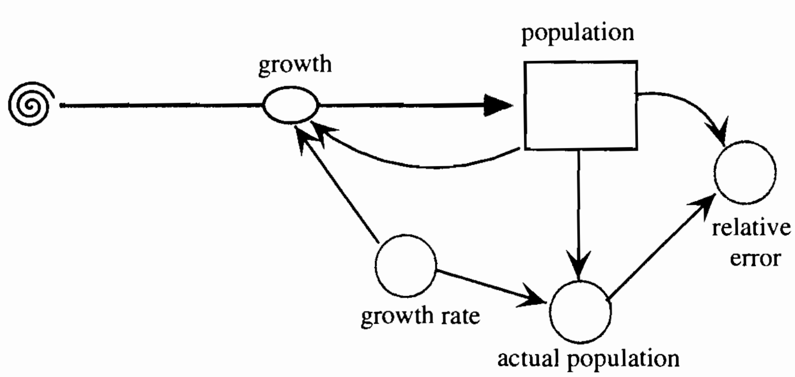

Figure 5.2.2 Unconstrained growth model with monitoring

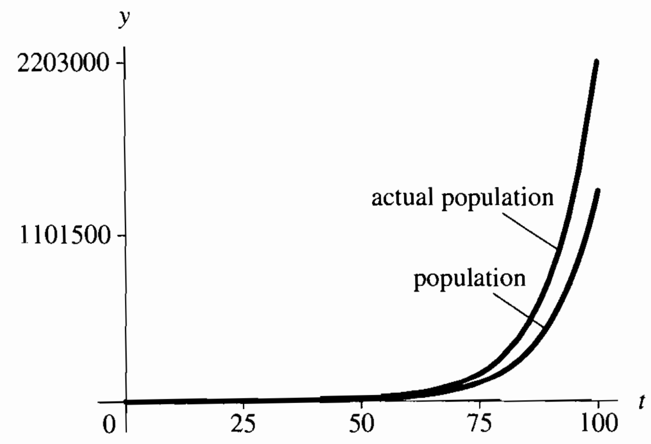

Figure 5.2.3 Graphs of analytical solution and Euler's Method solution with $\Delta t = 1$

Similarly, the converter *relative_error* computes the relative error as the absolute value (*ABS*) of the difference in Euler's estimate and the analytical population with the result divided by the latter, as follows:

```
ABS(population - actual_population)/actual_population
```

Figure 5.2.3 presents graphs of the analytical solution and the solution using Euler's Method with Δt being 1. As demonstrated, the simulated solution is below the analytical one. At the end of the run, at time 100, the analytical value for the population is 2,202,647, while the simulated solution using Euler's Method produces 1,378,061, so that the relative error is over 37.4%. For Δt being cut in half, the relative error is almost cut in half to 21.5% at time 100. The new simulated population is 1,729,258, which is considerably closer to the analytical solution of 2,202,647. If we cut the time step in half again so that Δt is 0.25, the relative also reduces by about half to 11.6% at time 100. Thus, the relative error is proportional to Δt. We say that the relative error is **on the order of Δt, O(Δt)**.

Quick Review Question 3

The analytical solution to $dP/dt = 10 + P/5$ with $P_0 = 500$ of Quick Review Question 1 is $P = 550e^{t/5} - 50$. Part b of that question showed that for $\Delta t = 0.1$ the Euler's Method estimate of P_1 is 511. Calculate the relative error as a percentage with four decimal places.

Of the three simulation techniques in this chapter, Euler's Method is the easiest to understand and has the fastest execution time but is the least accurate. We usually can reduce the error of the Euler Method by employing a smaller Δt, which unfortunately slows the simulation. Despite its shortcomings, the reasoning behind Euler's Method serves as an excellent introduction to the other techniques, Runge-Kutta 2 and Runge-Kutta 4, because each of these has Euler's Method embedded as the first step in its simulation.

Exercises

1. Use $dP/dt = 0.10P$ with $P_0 = 100$ and Euler's Method to calculate P_2 starting with $P_1 = 180$ at $t = 8$ in Figure 5.2.1.

In Exercises 2–5, for each differential equation with initial condition and Δt, calculate the following using Euler's Method and any other requested computation:

a. The first estimated point, such as P_1 where the differential equation is in terms of dP/dt

b. The second estimated point, such as P_2

2. $dP/dt = 0.10P$ With $P_0 = 100$ and $\Delta t = 2$
   c. The relative error for P_1
3. The logistic equation $dP/dt = 0.5(1 - P/1000)P$ with $P_0 = 20$ and $\Delta t = 2$
   c. The relative error for P_1, where the exact solution is $P(t) = \frac{10}{0.01+0.49e^{-0.5t}}$
4. The rate of change of the number of particles of radioactive carbon-14 in a dead tree $dA/dt = -2.783\ e^{-0.000121t}$ with $A_0 = 23{,}000$ particles and $\Delta t = 0.2$yr
   c. The relative error for A_2, where the exact solution is $A(t) = 23000\ e^{-0.000121t}$
5. The Gompertz differential equation $dN/dt = kN\ln(M/N)$ with $N(0) = 200$, $k = 0.1$, $M = 1000$, and $\Delta t = 0.5$
   c. The relative error for N_2, where the exact solution is $N(t) = Me^{\ln(N_0/M)e^{-kt}}$

Projects

Using Algorithm 2 for Euler's Method, develop a computational tool file to perform the simulations of Projects 1–7. Run the simulation for the indicated length of time and perform any other requested tasks.

1. Calculate P from $t = 0$ through $t = 100$, where $dP/dt = 0.10P$ with $P_0 = 100$ and $\Delta t = 2$. Calculate the relative error at each time step using the solution $P = 100e^{0.1t}$. Repeat the computation with $\Delta t = 0.25$. Check your results using a system dynamics tool. Use your results in a discussion of relative error.
2. Calculate P from $t = 0$ through $t = 100$ for logistic equation $dP/dt = 1.05(1 - P/1000)P$ with $P_0 = 500$ and $\Delta t = 2$. Calculate the relative error at each time step using the solution $P(t) = \frac{10}{0.01+0.49e^{-0.5t}}$. Repeat the computation with $\Delta t = 0.25$. Check your results using a system dynamics tool. Use your results in a discussion of relative error.
3. Suppose the instantaneous rate of change of the number of particles (A) of radioactive carbon-14 in a gram of a dead tree is $dA/dt = -2.783\ e^{-0.000121t}$ particles/year from the time t the tree dies with $A_0 = 23{,}000$ particles. Use Euler's Method to estimate of the total change in the number of particles of carbon-14 between years 10 and 20. Calculate the exact value of the definite integral with calculus or an appropriate computational tool and compute the relative error.

4. The Gompertz differential equation, which is one of the best models for predicting the growth of cancer tumors, follows:

$$\frac{dn}{dt} = KN\ln\left(\frac{M}{N}\right)$$

where N is the number of cancer cells. Calculate N from $t = 0$ through $t = 20$, where $k = 0.1$, $M = 1000$, and $\Delta t = 0.5$. Using the solution $N(t) = Me^{\ln(N_0/M)e^{-kt}}$ calculate the relative error at each time step. Repeat the computation with $\Delta t = 0.25$. Check your results using a system dynamics tool. Use your results in a discussion of relative error.

5. Estimate $\int_1^5 (-t^2 + 10t + 24)\,dt$ using $\Delta t = 0.25$. Calculate the exact value using calculus or an appropriate computational tool and compute the relative error of the simulated result.

6. Estimate $\frac{1}{\sqrt{2\pi}}\int_0^2 e^{-t^2}\,dt$ using $\Delta t = 0.1$. The corresponding indefinite integral does not exist. The function being integrated is the normal distribution with mean 0 and standard deviation 1.

7. Calculate $h(t)$ and $s(t)$ from $t = 0$ through $t = 50$ using $\Delta t = 0.25$ for the following system of differential equations:

$$\begin{cases} \dfrac{ds}{dt} = 2\,s - 0.02\,hs, & s_0 = 100 \\ \dfrac{dh}{dt} = 0.01\,sh - 1.06\,h, & h_0 = 15 \end{cases}$$

As Module 6.4, "Predator-Prey Model," discusses, this system is a model for predator (h) and prey (s) populations.

Answers to Quick Review Questions

1. **a.** 110 because $10 + (500)/5 = 110$
   b. 511 because $500 + 0.1(110) = 511$
2. **a.** t_n — **I.** Time at current time step
   b. t_0 — **F.** Initial time
   c. n — **H.** Number of current time step
   d. Δt — **A.** Change in time between time steps
   e. P_n — **D.** Estimated value of function at current time step
   f. P_{n-1} — **E.** Estimated value of function at previous time step
   g. $f(t_{n-1}, P_{n-1})$ — **C.** Derivative of function at estimated point for previous time step
   h. t_{n-1} — **J.** Time at previous time step
3. 0.0217% because

$$550e^{0.1/5} - 50 = 511.111 \quad \text{and}$$

$$|(511 - 511.111)|/511.111 = 0.000217 = 0.0217\%$$

References

Burden, Richard L., and J. Douglas Faires. 2001. *Numerical Analysis.* 7th ed. Pacific Grove, Calif.: Brooks/Cole Publishing Co.

Danby, J.M.A. 1997. *Computer Modeling: From Sports to Spaceflight . . . From Order to Chaos.* Richmond, Va.: Willmann-Bell.

Woolfson, M. M., and G. J. Pert. 1999. *An Introduction to Computer Simulation.* Oxford, U.K.: Oxford University Press.

Zill, Dennis G. 2001. *A First Course in Differential Equations with Modeling Applications.* 7th ed. Pacific Grove, Calif.: Brooks/Cole Publishing Co.

MODULE 5.3

Runge-Kutta 2 Method

Introduction

In Module 5.2, "Euler's Method," which is a prerequisite to the current module, we discuss the simplest of this chapter's simulation techniques for solving differential equations and computing definite integrals numerically. In this section, we consider a second and better simulation technique, **Euler's Predictor-Corrector (EPC) Method**, also called **Runge-Kutta 2.**

Euler's Estimate as a Predictor

In the current module, we consider the example of Module 5.2 on "Euler's Method," $dP/dt = 0.10P$ with $P_0 = 100$. As in that section, $f(t_n, P_n)$ is sometimes a more convenient notation for the derivative dP/dt at step n. Thus, at $(t, P) = (0, 100)$, the derivative is $f(0, 100) = 0.1(100) = 10$. According to that technique, using the slope of the tangent line at (t_{n-1}, P_{n-1}), we have the following computation for t_n and estimation of P_n:

$$t_n = t_0 + n\,\Delta t$$
$$P_n = P_{n-1} + f(t_{n-1}, P_{n-1})\,\Delta t$$

As Figure 5.2.1 of "Euler's Method" and Fig. 5.3.1 illustrates for $t_0 = 0$ and $\Delta t = 8$, the estimate at $t_1 = 8$ is the vertical coordinate of the point on the tangent line, $100 + 8(10) = 180$.

Corrector

To estimate (t_n, P_n), we would really like to use the slope of the chord from (t_{n-1}, P_{n-1}) to (t_n, P_n) instead of the slope of the tangent line at (t_{n-1}, P_{n-1}). As in Figure 5.3.2, if we know the slope of the chord between $(0, P(0)) = (0, 100)$ and

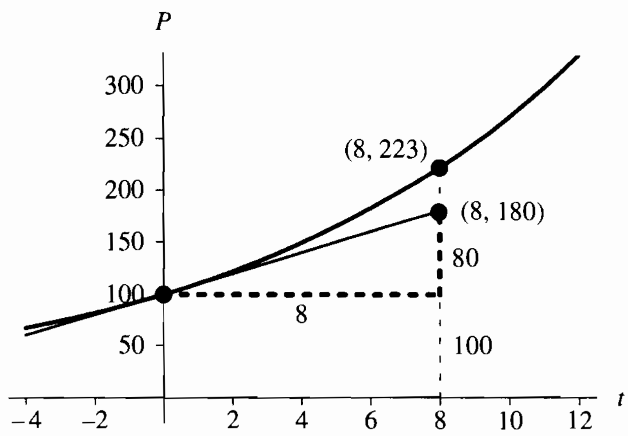

Figure 5.3.1 Actual point, $(8, P(8)) \approx (8, 223)$, and point obtained by Euler's Method, (8, 180)

$(8, P(8)) = (8, 100e^{0.10(8)}) \approx (8, 223)$ is approximately $\frac{223-100}{8-0} = \frac{123}{8}$, we can evaluate $P(8) = P(0) + slope\_of\_chord * \Delta t = 100 + \left(\frac{123}{8}\right)8 = 223$. However, to evaluate the slope of the chord, we must know $P(8) \approx 223$, which is the value we are trying to estimate. If we know the actual value, there is no need to estimate.

Although we do not know the slope of the chord between $(0, P(0))$ and $(8, P(8))$, we can estimate it as approximately the average of the slopes of the tangent lines at $P(0)$ and $P(8)$:

$$\begin{pmatrix} \text{slope of the chord} \\ \text{between (0, P(0)) and (8, P(8))} \end{pmatrix} \approx \frac{(\text{slope of tan at } P(0)) + (\text{slope of tan at } P(8))}{2}$$

Figure 5.3.3 depicts these two tangent lines.

How can we find the slope of the tangent line at $P(8)$ when we do not know $P(8)$? Instead of using the exact value, which we do not know, we predict $P(8)$ as in

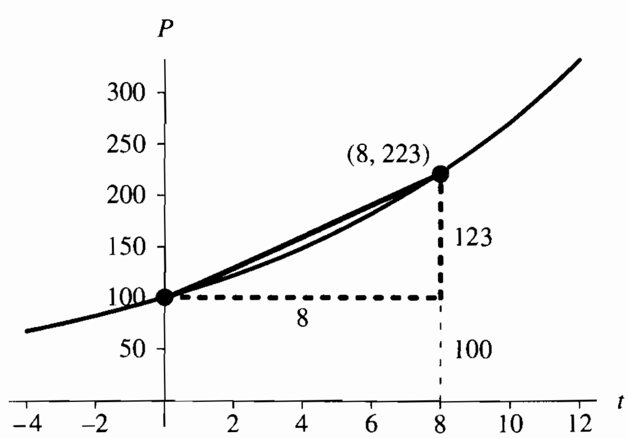

Figure 5.3.2 Actual point, $(8, P(8)) \approx (8, 223)$, along the chord between (0, 100) and (8, 223)

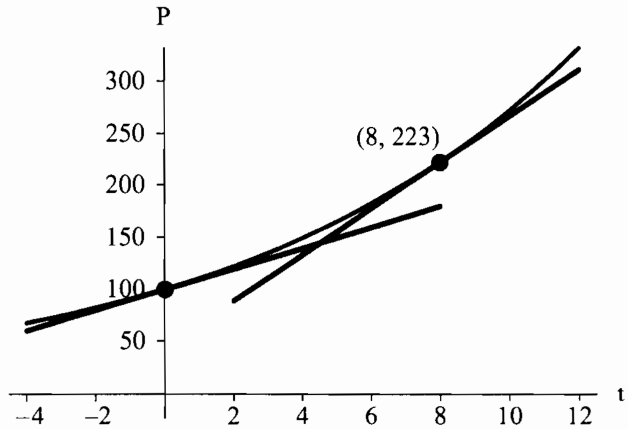

Figure 5.3.3 Tangent lines at (0, $P(0)$) and (8, $P(8)$)

Euler's Method. As the computation in the first section on "Euler's Estimate as a Predictor" shows, in this case, the prediction is $Y = 180$. We use the point (8, 180) in derivative formula to obtain an estimate of slope at $t = 8$. In this case, the slope of the tangent line at (8, 180), or the derivative, is $f(8, 180) = 0.1(180) = 18$. Using 18 as the approximate slope of the tangent line at (8, $P(8)$), we estimate the slope of chord between (0, $P(0)$) and (8, $P(8)$) as the following average of tangent line slopes:

$$\text{slope of chord} \approx (10 + 18)/2 = 0.5(10 + 18) = 14$$

As Figure 5.3.4 shows, using 14, the corrected estimate is $P_1 = 100 + 14(8) = 212$, which is closer to the actual value of 223.

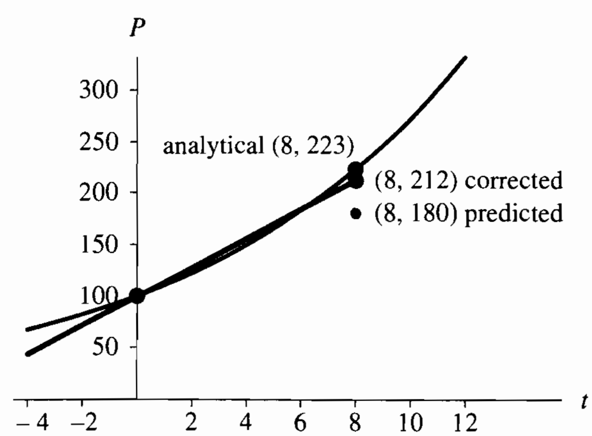

Figure 5.3.4 Predicted and corrected estimation of (8, $P(8)$)

Quick Review Question 1

Quick Review Question 1 of Module 5.2 on "Euler's Method" considered $dP/dt = 10 + P/5$ with $P_0 = 500$ and $\Delta t = 0.1$. Part a calculated the derivative at $t = 0$ to be 110, and Part b evaluated Euler's estimate of P_1 to be 511. Calculate the corrected estimate for P_1 using the technique of this section and decimal notation for your answer.

Runge-Kutta 2 Algorithm

Computations of the previous section illustrate Euler's Predictor-Corrector Method for estimating P numerically given a differential equation involving dP/dt. The algorithm for **Euler's Predictor-Corrector (EPC) Method**, or **Runge-Kutta 2**, is the same as Euler's with only one more statement in the loop to obtain the corrected value.

Algorithm for Euler's Predictor-Corrector (EPC) Method, or Runge-Kutta 2, with $f(t_{n-1}, P_{n-1})$ indicating the derivative dP/dt at step $n - 1$

> **Initialize t_0 and P_0**
> **Initialize *NumberOfSteps***
> **for n going from 1 to *NumberOfSteps* do the following:**
> $t_n = t_0 + n\,\Delta t$
> $Y_n = P_{n-1} + f(t_{n-1}, P_{n-1})\,\Delta t$, **which is the Euler's Method estimate**
> $P_n = P_{n-1} + 0.5\,(f(t_{n-1}, P_{n-1}) + f(t_n, Y_n))\,\Delta t$

Quick Review Question 2

Match each of the symbols below to its meaning in the Algorithm for Euler's Predictor-Corrector (EPC) Method. Here, "previous" means "immediately previous"; "EPC estimate" means "estimated value of function using the EPC Method"; and "Euler estimate" means "estimated value of function using Euler's Method."

- **A.** Average of derivatives of function at previous EPC estimate and current Euler estimate
- **B.** Average of derivatives of function at previous Euler estimate and current EPC estimate
- **C.** Derivative of function at EPC estimate for current time step
- **D.** Derivative of function at EPC estimate for previous time step
- **E.** Derivative of function at Euler estimate for current time step
- **F.** Derivative of function at Euler estimate for previous time step
- **G.** EPC estimate at current time step
- **H.** EPC estimate at previous time step

I. Euler estimate at current time step
J. Euler estimate at previous time step
a. Y_n
b. P_{n-1}
c. $f(t_{n-1}, P_{n-1})$
d. P_n
e. $f(t_n, Y_n)$
f. $0.5\,(f(t_{n-1}, P_{n-1}) + f(t_n, Y_n))$

Error

As noted above, the actual slope of the chord is $(P(8) - 100)/8 \approx (222.6 - 100)/8 \approx 15.3$, but 14 is certainly a better slope to use than 10 from Euler's Method. With Euler's Method, $P_1 = 180$, giving a relative error of $(180 - P(8))/P(8) \approx |180 - 222.6|/222.6 \approx 0.191 \approx 19.1\%$. We get a much better estimate with Euler's Predicator-Corrector (Runge-Kutta 2) Method, $P_1 = 212$, which has a relative error of $|212 - P(8)|/P(8) \approx |212 - 222.6|/222.6 \approx 0.047 \approx 4.7\%$.

As we saw in the "Error" section of Module 5.2, "Euler's Method," the relative error of Euler's method is on the order of Δt, $O(\Delta t)$. If we cut the time interval Δt in half, the relative error is halved as well. Using the same model with the Runge-Kutta 2 simulation method, Table 5.3.1 shows estimates of $P(100)$, whose actual value to 0 decimal places is 2,202,647, for $\Delta t = 1$, 0.5, and 0.25. As the time interval is cut by 1/2, the relative error is cut by about $(1/2)^2 = (1/4)$. Thus, the relative error of the EPC method is $O((\Delta t)^2)$, or on the order of $(\Delta t)^2$. Thus, although in each EPC algorithm iteration we must compute a correction, the EPC method is more accurate than Euler's Method.

Quick Review Question 3

The analytical solution to $dP/dt = 10 + P/5$ with $P_0 = 500$ of Quick Review Question 1 is $P = 550e^{t/5} - 50$. That question showed that the Euler's Predictor-Corrector Method estimate of P_1 is 511.11 for $\Delta t = 0.1$. Calculate the relative error as a percentage rounded to four decimal places.

Table 5.3.1
Estimates of $P(100)$ and Relative Errors for Various Changes in Time Using Runge-Kutta 2 Simulation Method, where $dP/dt = 0.10P$ with $P_0 = 100$

	EPC Estimates at Time 100	
Δt	*Estimated* $P(100)$	*Relative Error*
1.00	2,168,841	1.53%
0.50	2,193,824	0.40%
0.25	2,200,396	0.10%

Exercises

Repeat the exercises of Module 5.2 on "Euler's Method" using the Runge-Kutta 2 Method. Compare the relative errors with those of the corresponding exercises from Module 5.2.

Projects

Repeat the projects of Module 5.2 on "Euler's Method" using the Runge-Kutta 2 Method.

Answers to Quick Review Questions

1. 511.11 because $Y_1 = 511$; $f(0.1, 511) = 10 + 511/5 = 112.2$; $500 + (0.5)(110 + 112.2)(0.1) = 511.11$

2.
- **a.** Y_n — **I.** Euler estimate at current time step
- **b.** P_{n-1} — **H.** EPC estimate at previous time step
- **c.** $f(t_{n-1}, P_{n-1})$ — **D.** Derivative of function at EPC estimate for previous time step
- **d.** P_n — **G.** EPC estimate at current time step
- **e.** $f(t_n, Y_n)$ — **E.** Derivative of function at Euler estimate for current time step
- **f.** $0.5\,(f(t_{n-1}, P_{n-1}) + f(t_n, Y_n))$ — **A.** Average of derivatives of function at previous EPC estimate and current Euler estimate

3. 0.0002% because $550e^{0.1/5} - 50 = 511.111$ and $|511.11 - 511.111|/511.111 = 0.000002 = 0.0002\%$

References

Burden, Richard L., and J. Douglas Faires. 2001. *Numerical Analysis*. 7th ed. Pacific Grove, Calif.: Brooks/Cole Publishing Co.

Danby, J.M.A. 1997. *Computer Modeling: From Sports to Spaceflight . . . From Order to Chaos*. Richmond, Va.: Willmann-Bell.

Woolfson, M. M., and G. J. Pert. 1999. *An Introduction to Computer Simulation.* Oxford, U.K.: Oxford University Press.

Zill, Dennis G. 2001. *A First Course in Differential Equations with Modeling Applications*. 7th ed. Pacific Grove, Calif.: Brooks/Cole Publishing Co.