8.3 Numerical Methods: Runge-Kutta

We saw the improved Euler's method uses an average of the slope at the endpoints of each interval to get a better approximation for y_{n+1}. The Runge-Kutta method uses a weighted average of four slope values and is actually called the classic fourth order four-stage Runge-Kutta Method.

Using Runge-Kutta we have

$$y_{n+1} = y_n + h\,\frac{(k_{n1}+2\,k_{n2}+2\,k_{n3}+k_{n4})}{6}$$

where,

$$k_{n1} = f(t_n,\, y_n)$$
$$k_{n2} = f\left(t_n + \frac{1}{2}h,\, y_n + \frac{1}{2}h\,k_{n1}\right)$$
$$k_{n3} = f\left(t_n + \frac{1}{2}h,\, y_n + \frac{1}{2}h\,k_{n2}\right)$$
$$k_{n4} = f(t_n + h,\, y_n + h\,k_{n3})$$

Step 6 in the Euler method pseudo code would be:

Step 6. $k1 = f(t, y)$
$k2 = f\left(t + \frac{1}{2}h,\, y + \frac{1}{2}h * k1\right)$
$k3 = f\left(t + \frac{1}{2}h,\, y + \frac{1}{2}h * k2\right)$
$k4 = f(t + h,\, y + h * k3)$
$y = y + \left(\frac{h}{6}\right) * (k1 + 2\,k2 + 2\,k3 + k4)$
$t = t + h$

For the TI-84 program use A, B, C, and D for k1, k2, k3, and k4. One possible function (or module) in *Mathematica* is:

```
RK4[fn_, {t0_, y0_}, h_, n_] :=
 (*Create the function inicating the inputs*)
 Module[{tn, yn, k1, k2, k3, k4, table},      (*Define local variables for the module*)
  table = {{"n", "t", "y"}, {0, t0, y0}};     (*Create a table with heading
                                                and initial values*)
  {tn, yn} = {t0, y0};                        (*Set tn and yn to the initial values*)

  For[i = 1, i ≤ n, i++,                      (*Begin the loop*)
   k1 = fn /. {t → tn, y → yn};               (*Calculate all the k values*)
   k2 = fn /. {t → tn + 0.5 h, y → yn + 0.5 h k1};
   k3 = fn /. {t → tn + 0.5 h, y → yn + 0.5 h k2};
   k4 = fn /. {t → tn + h, y → yn + h k3};
   tn = tn + h;                               (*Increment tn*)
   yn = yn + h (k1 + 2 k2 + 2 k3 + k4)/6;     (*Increment yn*)

   AppendTo[table, {i, tn, yn}]               (*Append the new {tn,yn} to the table*)
  ];                                          (*Close the For loop*)
                                              (*Format and display the table*)
  Grid[table, Dividers → {{False, True, True}, {False, True}}, Alignment → "."]
 ]
```

Example 1 For the differential equation $y' = 1 - t + 4y$, $y(0) = 1$, compare the estimates of $y(2)$ using

a) the improved Euler's method with $h = 0.2$ and $h = 0.025$

b) Runge-Kutta with $h = 0.1$ and $h = 0.05$

Using the RK4 method above with $h = 0.1$ we get:

```
RK4[1 - t + 4 y, {0, 1}, 0.1, 20]
```

n	t	y
0	0	1
1	0.1	1.60893
2	0.2	2.50501
3	0.3	3.82941
4	0.4	5.79279
5	0.5	8.70932
6	0.6	13.0477
7	0.7	19.5071
8	0.8	29.1306
9	0.9	43.474
10	1.	64.8581
11	1.1	96.7453
12	1.2	144.3
13	1.3	215.227
14	1.4	321.019
15	1.5	478.819
16	1.6	714.203
17	1.7	1065.32
18	1.8	1589.08
19	1.9	2370.39
20	2.	3535.87

The actual value for $y(2)$ is

```
DSolve[{y'[t] == 1 - t + 4 y[t], y[0] == 1}, y[t], t] /. t → 2
```

$$\left\{\left\{y[2] \to \frac{1}{16}\left(5 + 19\, e^{8}\right)\right\}\right\}$$

```
N[%]
```

```
{{y[2.] → 3540.2}}
```

The error is only $\frac{3535.87-3540.2}{3540.2} \approx -0.00122$, or under by about 0.122%, not bad.