University of Oslo

MAT-INF3360

Introduction to Partial Differential Equations

Mandatory exercise 1

Author:

Thomas Larsen Greiner

Contents

1 Project 2.1

1.1 (a)

The following derivation is based on examples from Morten Hjorth-Jensen "Computational Physics". We first state that the area A under a curve $f(x)$ is given by the the definite integral

$$A = \int_a^b f(x)dx \tag{1}$$

This integral can be approximated by one trapezoid

$$A \approx \frac{f(a)+f(b)}{2}(b-a) \tag{2}$$

Where $f(a)$ and $f(b)$ are the values of the function f at points a and b. We can further divide a and b into smaller increments $x_0, x_1, x_2, ..., x_n, x_{n+1}$. We can then rewrite equation 2 as follows

$$\begin{aligned} A &\approx \frac{f(x_0)+f(x_1)}{2}(x_1-x_0)+\frac{f(x_1)+f(x_2)}{2}(x_2-x_1) \\ &\quad +\frac{f(x_2)+f(x_3)}{2}(x_3-x_2)+\ldots+\frac{f(x_n)+f(x_{n+1})}{2}(x_{n+1}-x_n) \end{aligned}$$

Where $f(x_0) = f(a)$ and $f(x_{n+1}) = f(b)$. If the intervals $(x_i - x_{i-1})$ are uniform we can write them as a increment h. We see that equation 2 can be written as follows

$$\begin{aligned} A &\approx \frac{h}{2}(f(x_0)+f(x_1)+f(x_1)+f(x_2)+f(x_2)+f(x_3)+...+f(x_n)+f(x_{n+1}) \\ &= \frac{h}{2}(f(a)+2f(x_1)+2f(x_2)+2f(x_3)+...2f(x_n)+f(b)) \\ &= \frac{h}{2}(f(a)+2\sum_{i=1}^{n} f(x_i)+f(b)) \end{aligned}$$

Where $h = (b-a)/(n+1)$ and $x_i = a + ih$ and $n > 0$. From equation 2 we see that we have repetition of the values in-between the end points $f(a)$ and $f(b)$. By multiplying 1/2 in will then give

$$A \approx h(\frac{f(a)}{2} + \sum_{i=1}^{n} f(x_i) + \frac{f(b)}{2}) \tag{3}$$

1.2 (b)

Procedure for numerical integration with the trapezoidal rule with a simple Matlab script:

```
clear;
a = 0;

b = 1;

n = 160; %test 10,20,40,80,160

h = (b - a)/(n + 1);

%fa = (a^5)/2;
%fb = (b^5)/2;

fa = (sqrt(abs(a - 0.5)))/2;
fb = (sqrt(abs(b - 0.5)))/2;

A = zeros(1,n);
x = zeros(1,n);

Asum = 0;

for i = 1:n;

    x(i) = a + i*h;

    %Insert function to integrate

    %A(i) = x(i)^5;
    A(i) = sqrt(abs(x(i) - 0.5));

    Asum = Asum + A(i);

end

TrapzA = (fa + Asum + fb)*h;
```

1.3 (c)

Considering $F(x) = x^5$. We have

$$A = \int_0^1 x^5 dx = \frac{1}{6}1^6 + \frac{1}{6}0^6 + C \tag{4}$$

Assuming $C = 0$, then

$$A \approx 0,16666... \tag{5}$$

From the Matlab computations

$n = 10 : A = 0,1701$
$n = 20 : A = 0,1676$
$n = 40 : A = 0,1669$
$n = 80 : A = 0,1667$
$n = 160 : A = 0,1667$

Considering $G(x) = \sqrt{|x - 0.5|}$

$$A = \int_0^1 \sqrt{|x - 0.5|} dx = \int_{u=-0.5}^{u=0.5} \sqrt{|u|} du \tag{6}$$

In this equation we have used integration by substitution with $u = x - 0.5$ and $du/dx = 1$. To get rid of the absolute value we define $|u|$ as follows

$$|u| = \begin{cases} u & \text{if } x \geq 0.5 \\ -u & \text{if } x \leq 0.5 \end{cases}$$

We can now split equation 6 into two integrals

$$A = \int_{u=0}^{u=0.5} \sqrt{u} du + \int_{u=-0.5}^{u=0} \sqrt{-u} du = 2\frac{2}{3}0.5^{\frac{3}{2}} + C \tag{7}$$

Assuming $C = 0$, then

$$A \approx 0,471405... \tag{8}$$

From the Matlab computations

$n = 10 : A = 0,4757$
$n = 20 : A = 0,4729$
$n = 40 : A = 0,4719$
$n = 80 : A = 0,4716$
$n = 160 : A = 0,4715$

1.4 (d)

From 2.54 we have

$$u(x) = x\int_0^1 (1-y)f(y)dy - \int_0^x (x-y)f(y)dy \tag{9}$$

By multiplying $f(y)dy$ into the parenthesis we get

$$u(x) = x\int_0^1 f(y)dy - x\int_0^1 yf(y)dy + x\int_0^x f(y)dy - \int_0^x yf(y)dy \tag{10}$$

We defined the functions

$$\alpha(x) = \int_0^x f(y)dy \text{ and } \beta(x) = \int_0^x yf(y)dy \tag{11}$$

which implies

$$\alpha(1) = \int_0^1 f(y)dy \text{ and } \beta(1) = \int_0^1 yf(y)dy \tag{12}$$

Substitution of the equations from (11) and (12) into 10 gives

$$u(x) = x\alpha(1) - x\beta(1) + x\alpha(x) - \beta(x) = x(\alpha(1) - \beta(1)) + x\alpha(x) - \beta(x) \tag{13}$$

1.5 (e)

This can be explained by the cumulative sum of the trapezoids. For one trapezoid we can write the area as follows

$$\alpha_1 = \frac{f(x_1) + f(x_0)}{2}h \approx \alpha(x_1) \tag{14}$$

and the next (cumulative) area as

$$\alpha_2 = \alpha_1 + \frac{f(x_2) + f(x_1)}{2}h \approx \alpha(x_2) \tag{15}$$

We can split the second term in equation 15 into two parts by doing a simple interpolation of the midpoint between $f(x_2)$ and $f(x_1)$, i.e splitting the trapezoids into two. We can in this case write equation 15 as

$$\alpha_2 = \alpha_1 + \frac{f(x_{1.5}) + f(x_1)}{2}(\frac{h}{2}) + \frac{f(x_2) + f(x_{1.5})}{2}(\frac{h}{2}) = \alpha_1 + \frac{h}{4}(f(x_1) + 2f(x_{1.5}) + f(x_2)) \tag{16}$$

which can be written generally as

$$\alpha_{i+1} = \alpha_i + \frac{h}{4}(f(x_i) + 2f(x_{i+0.5}) + f(x_{i+1})) \approx \alpha(x_i + 1) \tag{17}$$

The same derivation applies for β

$$\beta_2 = \beta_1 + \frac{x_2 f(x_2) + x_1 f(x_1)}{2}h \approx \beta(x_2) \tag{18}$$

$$\beta_2 = \beta_1 + \frac{x_{1.5} f(x_{1.5}) + x_1 f(x_1)}{2}(\frac{h}{2}) + \frac{x_2 f(x_2) + x_{1.5} f(x_{1.5})}{2}(\frac{h}{2}) \tag{19}$$

which gives

$$\beta_{i+1} = \beta_i + \frac{h}{4}(x_i f(x_i) + 2x_{i+0.5} f(x_{i+0.5}) + x_{i+1} f(x_{i+1})) \approx \beta(x_i + 1) \tag{20}$$

1.6 (f)

The result below are computed using the Matlab script presented in Figure 1.

```
clear

a = 0;
b = 1;
n = 100;
h = (b-a)/(n+1);

alpha = zeros(1,n+1);
beta = zeros(1,n+1);
f = zeros(1,n+1);
x = zeros(1,n+1);

alpha(1) = 0;
beta(1) = 0;
x(1) = 0;

for i = 1:n

    %insert function
    x(i+1) = a + i*h;

    %Exercise 2.1

    %f(1) = x(1)^2;f(i+1) = x(i+1)^2; %(a)
    %f(1) = exp(x(1));f(i+1) = exp(x(i+1)); %(b)

    f(1) = cos(x(1)); f(i+1) = cos(x(i+1)); %(c)

    %Exercise 2.1

    %f(1) = 1;
    %f(i+1) = 1;

    %Exercise 2.2
    %f(1) = x(1);
    %f(i+1) = x(i+1);

    fmean = (f(i+1)+ f(i))/2;
    xmean = (x(i+1) + x(i))/2;

    alpha(i+1) = alpha(i) + (h/4)*(f(i) + 2*fmean + f(i+1));

    beta(i+1) = beta(i) + (h/4)*(x(i)*f(i) + 2*xmean*fmean + x(i+1)*f(i+1));

end

u = zeros(1,n);
u(1) = 0;
u(2) = 0;

for j = 2:n

    u(j+1) = x(j)*(alpha(end) - beta(end)) + beta(j) - x(j)*alpha(j);

end

figure
plot(x,u,'o','LineWidth',1.5)
grid on

%%
%test exact

xx = 0:0.0001:1;

%ux = (1/12)*xx.*(1 - xx.^3); %(a)
%ux = -1*exp(xx) + xx.*(exp(1) - 1) + 1; %(b)
ux = cos(xx) - 1 + xx - cos(1)*xx; %(c)

%ux = 0.5*xx.*(1 - xx);

%ux = (1/6)*xx.*(1 - xx.^2);

%figure
hold on
plot(xx,ux,'r','LineWidth',3)
hold off
legend('Approximation','Exact')
```

Figure 1: Matlab script for numerical integration

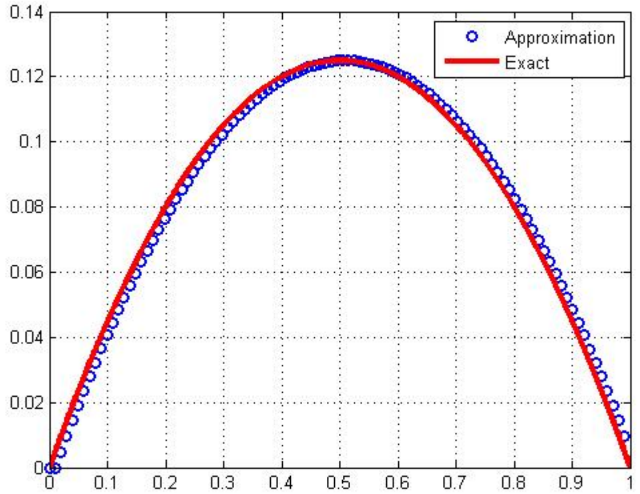

Figure 2: Numerical simulation results for Example 2.1 by using n=100

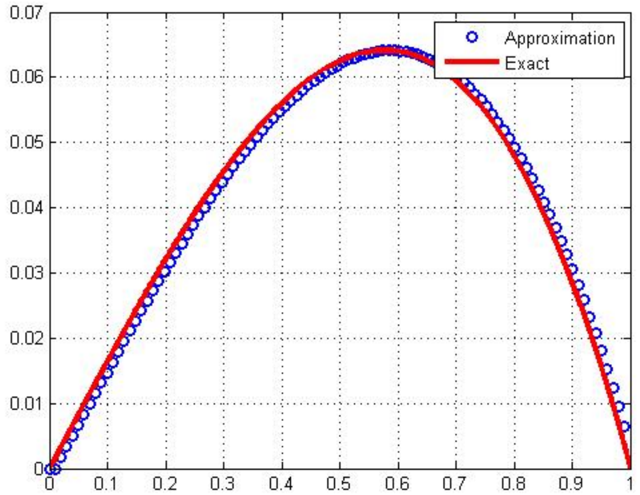

Figure 3: Numerical simulation results for Example 2.2 by using n=100

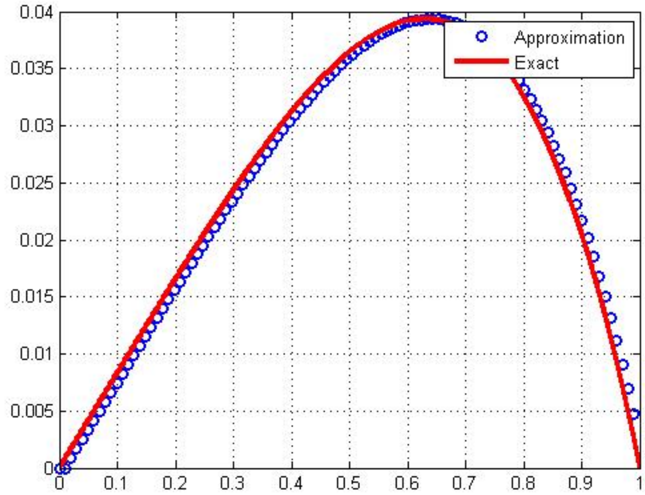

Figure 4: Numerical simulation results for Exercise 2.2 (a) by using n=100

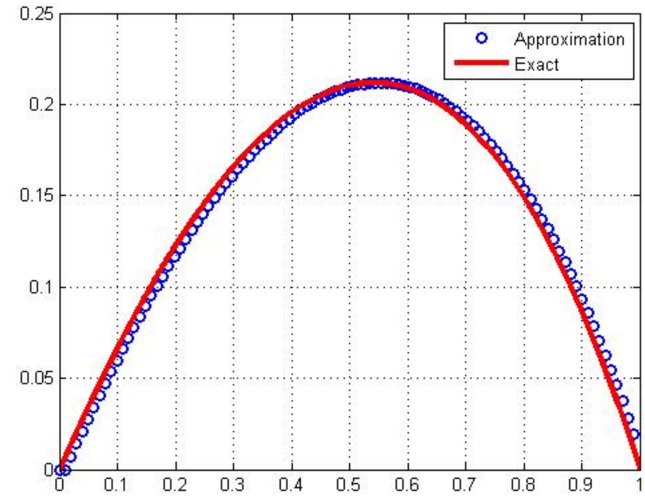

Figure 5: Numerical simulation results for Exercise 2.2 (b) by using n=100

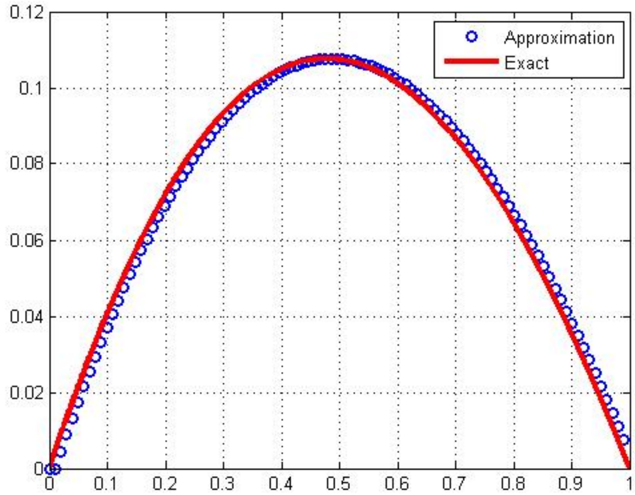

Figure 6: Numerical simulation results for Exercise 2.2 (c) by using n=100

The accuracy of the approximation increases with larger value of n, as it did with the trapezoidal rule. However not tested, I would assume this approximation to converge faster as we interpolate between trapezoids, which could give greater accuracy of the approximation.

1.7 (g)

Instead of increasing the value of n, we can instead use methods of higher order of accuracy, like the $Simpson'srule$

$$A \approx \frac{h}{3}(f(a) + f(b) + 2\sum_{i=1}^{n-1} f(x_{2i}) + 4\sum_{i=1}^{n} f(x_{2i-1})) \tag{21}$$

As we see in equation (18) we do not increase n, but reduces the error of the integration.