

Numerical Analysis

Assignment 13

Chuan Lu

December 4, 2017

Problem 1. Problem 5.23

(a). Integrate (5.4.2) with respect to x on $[0, 1]$, we get from the convergence of the function series,

$$\int_0^1 \left(\frac{t}{e^t - 1} e^{tx} - \frac{t}{e^t - 1} \right) dx = \sum_{j=1}^{\infty} \frac{t^j}{j!} \int_0^1 B_j(x) dx,$$

thus

$$1 - \frac{t}{e^t - 1} = \sum_{j=1}^{\infty} \frac{t^j}{j!} \int_0^1 B_j(x) dx,$$

with Taylor expansion of the left term of (5.4.5), we have $B_0 = 1$, thus

$$\sum_{j=1}^{\infty} (B_j + \int_0^1 B_j(x) dx) \frac{t^j}{j!} = 0.$$

The left term can be regarded as a Taylor series of a function f at $t = 0$ which satisfies $f(0) = 0$. By the uniqueness of Taylor expansion of a analytic function, we know $f \equiv 0$. Then for all $j > 0$,

$$B_j + \int_0^1 B_j(x) dx = 0.$$

Now we know $B_0 = 1$, $B_1 = -\frac{1}{2}$, thus from definition (5.4.5),

$$\frac{t}{e^t - 1} + \frac{1}{2}t = 1 + \sum_{j=2}^{\infty} B_j \frac{t^j}{j!} = \frac{t e^t + 1}{2 e^t - 1} \equiv g(t).$$

We have

$$g(-t) = -\frac{t e^{-t} + 1}{2 e^{-t} - 1} = -\frac{t}{2} \frac{1 + e^t}{1 - e^t} = g(t),$$

hence $g(t)$ is an even function. Substitute $t = -t$ into last term,

$$1 + \sum_{j=2}^{\infty} (-1)^j B_j \frac{t^j}{j!} = g(-t) = g(t) = 1 + \sum_{j=2}^{\infty} B_j \frac{t^j}{j!}.$$

By simplification,

$$\sum_{j=1}^{\infty} B_{2j+1} \frac{t^{2j+1}}{(2j+1)!} = 0.$$

With the same arguments and the uniqueness of Taylor expansion, we have

$$B_{2j+1} = 0, \text{ for all } j \geq 1.$$

(b). Take derivatives respective to x on both sides of (5.4.2),

$$\frac{t^2}{e^t - 1} e^{tx} = \sum_{j=1}^{\infty} B'_j(x) \frac{t^j}{j!}.$$

Hence,

$$\sum_{j=1}^{\infty} B'_j(x) \frac{t^j}{j!} = t \frac{t(e^{tx} - 1)}{e^t - 1} + \frac{t^2}{e^t - 1} = \sum_{j=1}^{\infty} B_j(x) \frac{t^{j+1}}{j!} + \sum_{j=0}^{\infty} B_j \frac{t^{j+1}}{j!} = \sum_{j=2}^{\infty} j B_{j-1}(x) \frac{t^j}{j!} + \sum_{j=1}^{\infty} j B_{j-1} \frac{t^j}{j!}$$

we have

$$tB'_1(x) + \sum_{j=2}^{\infty} B'_j(x) \frac{t^j}{j!} = B_0 t + \sum_{j=2}^{\infty} j(B_{j-1}(x) + B_{j-1}) \frac{t^j}{j!}.$$

Since $tB'_1(x) = B_0 t$, with the uniqueness of Taylor expansion,

$$B'_j(x) = j(B'_{j-1}(x) + B_{j-1}), \text{ for all } j \geq 2.$$

As proved in (a), $B_{2j+1} = 0$ for all $j \geq 1$, the conclusion holds.

Problem 2. Problem 5.31

```

1 function T = romberg(f, a, b, n)
2 % Romberg extrapolation
3 % f: function
4 % x: interval: [a, b]
5 % n: order
6 T = zeros(n, n);
7 T(1, 1) = (feval(f, a)+feval(f, b))/2*(b-a);
8
9 for i = 2:n
10     k = 2^(i-1);
11     h = (b-a)/k;
12     eval_points = a + (2*linspace(1, k, k)-1)*h/2;
13     sums = sum(feval(f, eval_points));
14     T(i, 1) = 0.5*(h*sums + T(i-1, 1));
15     m = 1;
16     for j = 2:i
17         m = m*4;
18         T(i, j) = (m*T(i, j-1)-T(i-1, j-1))/(m-1);
19     end
20 end
21
22 T = diag(T);

```

```

1 format long g;
2
3 f1 = @(x) exp(-x.^2);
4 f2 = @(x) x.^2.5;
5 f3 = @(x) 1./(1+x.^2);
6 f4 = @(x) 1./(2+cos(x));
7 f5 = @(x) exp(x).*cos(4*x);
8
9 a1 = 0; b1 = 1;
10 a2 = 0; b2 = 1;
11 a3 = -4; b3 = 4;
12 a4 = 0; b4 = 2*pi;
13 a5 = 0; b5 = pi;
14
15 N = 20;

```

```

16 T1 = romberg(f1, a1, b1, N);
17 T2 = romberg(f2, a2, b2, N);
18 T3 = romberg(f3, a3, b3, N);
19 T4 = romberg(f4, a4, b4, N);
20 T5 = romberg(f5, a5, b5, N);
21
22 res = [T1 T2 T3 T4 T5];
23
24 dlmwrite('res2.m', res, 'delimiter', ' ', 'precision', '%2.9f'
);

```

```

1 0.683939721 0.500000000 0.470588235 2.094395102 37.920111314
2 0.731045203 0.339463097 1.223529412 2.792526803 -0.705452807
3 0.739446087 0.310849407 1.993202614 3.297841177 3.229586000
4 0.743193648 0.298084229 2.365757648 3.468111523 2.177291245
5 0.745016009 0.291875140 2.508348805 3.547746390 1.737097813
6 0.745920955 0.288791713 2.579942644 3.587716961 1.519545872
7 0.746372654 0.287252625 2.615800086 3.607662729 1.410943245
8 0.746598407 0.286483408 2.633718795 3.617631337 1.356665152
9 0.746711272 0.286098841 2.642677198 3.622615109 1.329529004
10 0.746767702 0.285906563 2.647156280 3.625106928 1.315961292
11 0.746795918 0.285810424 2.649395806 3.626352829 1.309177482
12 0.746810025 0.285762355 2.650515567 3.626975779 1.305785582
13 0.746817079 0.285738320 2.651075447 3.627287254 1.304089633
14 0.746820606 0.285726303 2.651355387 3.627442991 1.303241659
15 0.746822369 0.285720294 2.651495357 3.627520860 1.302817671
16 0.746823251 0.285717290 2.651565342 3.627559794 1.302605678
17 0.746823692 0.285715788 2.651600335 3.627579261 1.302499681
18 0.746823912 0.285715037 2.651617831 3.627588995 1.302446683
19 0.746824023 0.285714661 2.651626579 3.627593862 1.302420183
20 0.746824078 0.285714473 2.651630953 3.627596295 1.302406934

```

Cmp. When comparing with Simpson's rule and trapezoidal rule, we can find that romberg extrapolation converges quicker than the corresponding order of the two rules, but slower than the 2^N order of the two rules.

Problem 3. Problem 5.37

```

1 function res = gauss_laguerre(f, n)
2 % Gauss-Laguerre Quadrature
3 % f: func
4 % n: num of points
5
6 %%% Reference: MATLAB documentation %%%
7 syms t;
8 w(t) = exp(-t);
9 F = laguerreL(n, t);
10 x = vpasolve(F);
11
12 y = sym('y', [n, 1]);
13 sys = sym(zeros(n));
14 for k = 0:n-1
15     sys(k+1) = sum(y.*(x.^k)) == int(t^k*w(t), t, 0, inf);

```

```

16 end
17 a = vpasolve(sys, y);
18 a = structfun(@double, a);
19 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
20 res = sum(a.*f(x));

```

Sol. (a) By (5.6.11),

$$\int_0^\infty e^{-x^2} dx = \int_0^\infty e^{-x} e^{x-x^2} dx.$$

(b) By (5.6.11),

$$\int_0^\infty \frac{x}{(1+x^2)^2} dx = \int_0^\infty e^{-x} \frac{x e^x}{(1+x^2)^2} dx.$$

(c) By (5.6.11),

$$\int_0^\infty \frac{\sin x}{x} dx = \int_0^\infty e^{-x} \frac{e^x \sin x}{x} dx.$$

The result is as follows, from left to right are (a), (b), (c); the first part is the result, and the second part is the error.

1	1.0879867064	0.5927453354	1.0961083700
2	0.8476788354	0.4816145789	1.2060830918
3	0.8791464499	0.4870482698	1.1347650995
4	0.8925955133	0.4974949629	1.0269639487

1	0.1653274309	0.0927453354	0.4746879568
2	0.4056353019	0.0183854211	0.3647132350
3	0.3741676874	0.0129517302	0.4360312273
4	0.3607186240	0.0025050371	0.5438323781

Problem 4. Problem 5.40