MATH 128A: Homework #5

Professor John Strain
Assignment: 1-8

Eric Chuu

July 28, 2016

Problem 1

(a) Find an exact formula for the cubic polynomial $P_3(x) = x^3 + \cdots$ such that

$$\int_{-1}^{1} P_3(x)q(x)dx = 0 \tag{1}$$

for any quadratic polynomial q.
(b) Find exact formulas for the three roots x_1, x_2, x_3 of the equation $P_3(x) = 0$.
(c) Find the exact formulas for the integration weights w_1, w_2, w_3 such that

$$\int_{-1}^{1} q(x)dx = \sum_{j=1}^{3} w_i q(x_j) \tag{2}$$

exactly whenever q is a polynomial of degree 5.
(d) Given any real numbers $a < b$, find exact formulas for points $y_j \in [a, b]$ and weights $u_j > 0$ such that

$$\int_{a}^{b} q(x)dx = \sum_{j=1}^{3} u_j q(y_j) \tag{3}$$

whenever q is a polynomial of degree 5.
(e) Explain why each of the three factors in the error estimate

$$\int_{a}^{b} f(x)dx - \sum_{j=1}^{3} u_j f(y_j) = C_6 f^{(6)}(\xi) \int_{a}^{b} (y - y_1)^2 (y - y_2)^2 (y - y_3)^2 dy \tag{4}$$

is inevitable and determine the exact value of the constant C_6.
(f) Use your code `gadap.m` from problem 2 to evaluate

$$E_6 = \int_{-1}^{1} (x - x_1)^2 (x - x_2)^2 (x - x_3)^2 dx \tag{5}$$

to 3-digit accuracy.

Solution

(a) We can use the Gram-Schmidt Orthogonalization process on the interval $(-1, 1)$, along with the first three Legendre polynomials to get an exact formula for the cubic polynomial $P_3(x)$

$$P_3(x) = \left(x - \frac{\int_{-1}^{1} x(x^2 - \frac{1}{3})^2 dx}{\int_{-1}^{1} (x - \frac{1}{3})^2} ddx \right) \left(x^2 - \frac{1}{3} \right) - \left(\frac{\int_{-1}^{1} (x^2 - \frac{1}{3})^2 dx}{\int_{-1}^{1} x^2 dx} \right) x$$

$$\boxed{P_3(x) = x^3 - \frac{3}{5} x}$$

(b) Using the above formula for $P_3(x)$, we can find the exact formula for the roots as well

$$x^2 \left(x - \frac{3}{5} \right) = 0$$

which implies that $x_1 = -\sqrt{\frac{3}{5}}, x_2 = 0, x_3 = \sqrt{\frac{3}{5}}$.

(c) Using the Lagrange Basis polynomials as the integration weights, we get

$$w_1 = \int_{-1}^{1} L_1(x) = \int_{-1}^{1} \frac{(x-0)\left(x - \sqrt{3/5}\right)}{\left(\sqrt{3/5} - 0\right)\left(-\sqrt{3/5} - \sqrt{3/5}\right)} dx = \frac{5}{9}$$

$$w_2 = \int_{-1}^{1} L_2(x) = \int_{-1}^{1} \frac{(x + \sqrt{3/5})\left(x - \sqrt{3/5}\right)}{\left(0 + \sqrt{3/5}\right)\left(0 - \sqrt{3/5}\right)} dx = \frac{8}{9}$$

$$w_2 = \int_{-1}^{1} L_2(x) = \int_{-1}^{1} \frac{(x + \sqrt{3/5})(x - 0)}{\left(\sqrt{3/5} + \sqrt{3/5}\right)\left(\sqrt{3/5} - 0\right)} dx = \frac{8}{9}$$

(d) We can rescale the interval to $[a, b]$ so that $y_j = \frac{b-a}{2} x_j + \frac{a-b}{2}$, which gives us the points y_j, $1 \leq j \leq 3$: $y_1 = -\sqrt{3/5}(b-a)/2 + (a+b)/2$, $y_2 = (a+b)/2$, and $y_3 = \sqrt{3/5}(b-a)/2 + (a+b)/2$. Then we calculate u_1, u_2, u_3 like before, using the Lagrange basis polynomials

$$u_1 = \int_a^b L_1(x)dx = \int_a^b \frac{5}{6}\left(x^2 - x\sqrt{3/5}\right) dx = \frac{1}{36}(-10a^3 + 3\sqrt{15}a^2 + 10b^2 - 3\sqrt{15}b)$$

$$u_2 = \int_a^b L_2(x)dx = \int_a^b \frac{-5}{3}\left(x^2 - 3/5\right) dx = \frac{5a^3}{9} - a - \frac{5b^3}{9} + b$$

$$u_3 = \int_a^b L_3(x)dx = \int_a^b \frac{5}{6}\left(x^2 + x\sqrt{3/5}\right) dx = \frac{1}{36}(-10a^3 - 3\sqrt{15}a^2 + 10b^2 + 3\sqrt{15}b)$$

(e) Looking at the factors on the right hand side of equation (4) above, we see that the integral is inevitable because $f(x) = f(y_j)$ for $j = 1, 2, 3$, so the error vanishes at these points. The sixth derivative term is inevitable because a polynomial of degree 5 will vanish upon taking its sixth derivative. Finally, $C_6 = \frac{1}{6!}$ is inevitable because the it must cancel with the constants that result from taking 6 derivatives.

(f) Using `gadap.m` to evaluate the integral in (5) to 3-digit accuracy, we get

$$\texttt{int} = 0.0457$$
$$n = 18$$

where `int` is the integral approximation and n is the number of function evaluations. □

Problem 2

(a) Write, test, and debug an adaptive 3-point Gaussian integration code `gadap.m`. Build a list `abt` $= [a_1, b_1, t_1], \ldots, [a_m.b_n, t_n]$ of n intervals $[a_j, b_j]$ and approximate integrals $t_j \approx \int_{a_j}^{b_j} f(x)dx$, computed with 3-point Gaussian integration from problem 1.

(b) Approximate the integral $\int_0^1 x^{-x}dx$ with your code from (a). Measure total number of function evaluations required to obtain 12-digit accuracy. Plot the accepted intervals. Compare the results with those obtained in previous problem set by Romberg integration.

Solution

The code for `gadap.m` is on the following page. We also keep track of the number of function evaluations throughout the calculation. When using `gadap.m` to approximate the integral $\int_0^1 x^{-x}dx$ to 6-digit accuracy, we get

$$\texttt{int} = 1.291286113679926$$
$$n = 78$$

where `int` is the approximation for the integral, and `n` is the number of function evaluations. Compared to Romberg Integration, which needed more than a million function evaluations, adaptive 3-point Gaussian clearly converges to the solution more quickly. If we evaluate $\int_0^1 x^{-x}dx$ to 12-digit accuracy, we find that we need

$$\texttt{int} = 1.291285997062738$$
$$n = 798$$

which is still considerably smaller than 10^6. □

```
% a, b: interval endpoints with a < b
% f: hardcoded to evaluate x^(-x)
% tol: User-provided tolerance for integral accuracy
% int: approximation to the integral
% evals: number of function evaluations

function [int, evals] = gadap (a, b, TOL)
    h = ( b - a ) / 2;
    f0 = f(a + h - sqrt(3 / 5) * h );
    f1 = f(a + h);
    f2 = f(a + h + sqrt(3 / 5) * h);
    ff = h * ( 5 * f0 + 8 * f1 + 5 * f2 ) / 9;

    [approx, error, function_evals] = g(ff, a, b, TOL);

    int = approx;
    evals = function_evals;
    return;

    function [approx, error, evals] = g( ff, a, b, TOL )
        h = ( b - a ) / 4;
        f1 = f(a + h - sqrt(3 / 5) * h);
        f2 = f(a + h);
        f3 = f(a + h + sqrt(3 / 5) * h);
        f4 = f(b - h - sqrt(3 / 5) * h);
        f5 = f(b - h);
        f6 = f(b - h + sqrt(3 / 5) * h);
        ff1 = h * ( 5 * f1 + 8 * f2 + 5 * f3 ) / 9;
        ff2 = h * ( 5 * f4 + 8 * f5 + 5 * f6 ) / 9;

        err_est = abs ( ff - ff1 - ff2 );
        if ( err_est < (42.0 * TOL) )
           approx = ff1 + ff2;
           error = err_est / 42.0;
           evals = 6;
           return;
        else
           [a1, e1, n1] = g( ff1, a, a + 2 * h, TOL / 2.0 );
           [a2, e2, n2] = g( ff2, a + 2 * h, b, TOL / 2.0 );
           approx = a1 + a2;
           error = e1 + e2;
           evals = n1 + n2 + 6;
           return;
        end;

    % function to integrate
    function y = f(x)
        y = x^(-x);
```

Problem 3

Implement a MATLAB function `pleg.m` of the form

```
function [ p, pp, ppp ] = pleg(x,n )
% x:  evaluation point
% n:  degree of polynomial
```

This function evaluates a Legendre polynomial P_n of degree n, with its derivatives P'_n and P''_n, at an evaluation point x with $|x| \le 1$. Here $P_0 = 1, P_1(x) = x$ and P_n, P'_n, and P''_n are determined by the recurrence

$$\begin{aligned} P_n(x) &= xP_{n-1}(x) - c_{n-1}P_{n-2}(x) \\ P'_n(x) &= P_{n-1}(x) + xP'_{n-1}(x) - c_{n-1}P'_{n-2}(x) \\ P''_n(x) &= 2P'_{n-1} + xP''_{n-1}(x) - c_{n-1}P''_{n-2}(x), \end{aligned}$$

for $n \ge 2$ where $c_n = n^2/(4n^2 - 1)$.

Solution

```
% this function evaluates Legendre polynomial P_n
% of degree n, with its derivatives P'_n and P"_n
% at an evaluation point x with |x| <= 1

% x = evaluation point
% n = degree of polynomial

function [p, pp, ppp] = pleg(x, n)
    P = zeros(n+1, 1); % store function evaluations
    P(1,1) = 1;
    P(2,1) = x;

    PP = zeros(n+1, 1); % store 1st derivative evals
    PP(1,1) = 0;
    PP(2,1) = 1;

    PPP = zeros(n+1, 1); % store 2nd derivatives evals
    PPP(1,1) = 0;
    PPP(2,1) = 0;

    for i = 3:(n+1)
        c = (i - 2)^2 / (4 * (i - 2)^2 - 1);
        P(i, 1)   = x * P(i - 1, 1) - c * P(i - 2, 1);
        PP(i, 1)  = P(i - 1, 1) + x * PP(i - 1) - c * PP(i - 2);
        PPP(i, 1) = 2 * PP(i - 1, 1) + x * PPP(i - 1, 1) - c * PPP(i - 2, 1);
    end

    p   = P(n + 1, 1);
    pp  = PP(n + 1, 1);
    ppp = PPP(n + 1, 1);
```

□

Problem 5

(a) For arbitrary $s \in \mathbb{R}$, find the exact solution of the initial value problem

$$y'(t) = \frac{1}{2}(y(t) + y(t)^3) \tag{6}$$

with $y(0) = s > 0$.

(b) Show that the solution blows up when $t = \ln(1 + 1/s^2)$.

Solution

(a) We recognize that y is a function of t, and we omit t in the calculations until we use the initial condition for convenience. We can rearrange the differential equation and use partial fractions to see that

$$\begin{aligned} \frac{y'}{y(1+y^2)} &= \frac{1}{2} \\ \frac{b}{y} + \frac{cy}{1+y^2} &= \frac{1}{2} \\ \frac{b(1+y^2) + cy}{y(1+y^2)} &= \frac{1}{2} \end{aligned}$$

when $b = 1, c = -1$. Plugging these values back into the third equality, we get

$$\begin{aligned} \frac{1}{y} - \frac{y}{1+y^2} &= \frac{1}{2} \\ \int \frac{1}{2} dt &= \int \frac{1}{y} dy - \int \frac{y}{1+y^2} dy \\ \frac{t}{2} &= \ln(y) - \frac{1}{2}\ln(1+y^2) + c \end{aligned}$$

for some $c \in \mathbb{R}$. Using the initial condition, $y(0) = s > 0$, we get

$$\begin{aligned} c &= -\ln(s) + \frac{1}{2}\ln(1+s^2) \\ \frac{t}{2} &= \ln(y) - \ln(s) - \frac{1}{2}\ln(1+y^2) + \frac{1}{2}\ln(1+s^2) \\ &\boxed{t = \ln\left(\frac{y^2}{s^2} \cdot \frac{1+s^2}{1+y^2}\right)} \end{aligned} \tag{7}$$

(b) We define

$$z := \frac{y^2}{1+y^2} = e^t\left(\frac{s^2}{1+s^2}\right)$$

Note this holds since $y(0) = s$, and the exponential in the last equality reduces to 1. Then

$$y^2 = (1+y^2)z \implies y^2 = \frac{z}{1-z}$$

since $1 - z = 1 - y^2/1 + y^2 = 1/(1+y^2)$. Using this equality, we can further evaluate,

$$y^2 = \frac{e^t\left(\frac{s^2}{1+s^2}\right)}{1-z} = \frac{e^t\left(\frac{s^2}{1+s^2}\right)}{1 - e^t\left(\frac{s^2}{1+s^2}\right)}$$

Note that when $t = \ln(1 + \frac{1}{s^2})$, the denominator of the above equality evaluates to

$$1 - \left(1 + \frac{1}{s^2}\right) \cdot \left(\frac{s^2}{1+s^2}\right) = 1 - 1 = 0$$

so $y \to \infty$ and the solution explodes. $\square$

Problem 6

(a) Find the general solution of the difference equation

$$u_{j+2} = u_{j+1} + u_j \tag{8}$$

(b) Find all initial values u_0 and u_1 such that u_j remains bounded by a constant as $j \to \infty$.

Solution
(a) We can use the recurrence relation given in (8) to generalize the difference equation

$$\begin{bmatrix} u_{j+1} \\ u_{j+2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} u_j \\ u_{j+1} \end{bmatrix}$$

We define the following matrices:

$$U_{j+1} = \begin{bmatrix} u_{j+1} \\ u_{j+2} \end{bmatrix} \qquad F = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \qquad U_j = \begin{bmatrix} u_j \\ u_{j+1} \end{bmatrix}$$

Then using this recurrence relation, we can reduce U_j down the matrix with elements u_0, u_1,

$$U_{j+1} = F^{j+1} \begin{bmatrix} u_0 \\ u_1 \end{bmatrix}$$

Since F is symmetric, it is diagonalizable, so upon diagonalizing and noticing that the inner inverses in the matrices raised to the j-th power cancel, we find a general solution for the recurrence:

$$\boxed{U_j = SD^j S^{-1} U_0}$$

where

$$S = \begin{bmatrix} \frac{-1+\sqrt{5}}{2} & \frac{-1-\sqrt{5}}{2} \\ 1 & 1 \end{bmatrix} \qquad D = \begin{bmatrix} \frac{1+\sqrt{5}}{2} & 0 \\ 0 & \frac{1-\sqrt{5}}{2} \end{bmatrix} \qquad S^{-1} = \begin{bmatrix} \frac{1}{\sqrt{5}} & \frac{1+\sqrt{5}}{2\sqrt{5}} \\ \frac{-1}{\sqrt{5}} & \frac{-1+\sqrt{5}}{2\sqrt{5}} \end{bmatrix} \qquad U_0 = \begin{bmatrix} u_0 \\ u_1 \end{bmatrix}$$

(b) For numbers c_1, c_2 that can be calculated by means of the general solution above, we have that

$$\begin{aligned} u_0 &= c_1 + c_2 \\ u_1 &= c_1\lambda_1 + c_2\lambda_2 \end{aligned}$$

where λ_1, λ_2 are the entries along the diagonal of the matrix D above. Note that $|\lambda_1| > 1$ and $|\lambda_2| < 1$. When we take numbers greater than 1 to the j-th power as $j \to \infty$, the numbers similarly go to ∞. When number less than 1 are taken the the j-th power as $j \to \infty$, the numbers go to 0. Thus, we want to exclude λ_1 from the above equalities, and we can do so by requiring $u_0 = c_2$ and $u_1 = \lambda_2 c_2$, which gives us

$$\begin{bmatrix} u_0 \\ u_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ \lambda_1 & \lambda_2 \end{bmatrix} \begin{bmatrix} 0 \\ c_2 \end{bmatrix}$$

With these initial values of u_1, u_2, we ensure that u_j remains bounded as $j \to \infty$. □

Problem 7

(a) Write, test and debug a MATLAB function of the form

```
function u = euler(a, b, ya, f, n)
% a,b:  interval endpoints with a < b
% n:  number of steps with h = (b-a)/n
% ya:  vector y(a) of initial conditions
% f:  function handle f(t,y) to integrate (y is a vector)
% u:  output approximation to the final solution vector y(b)
```

which approximates the final solution vector $y(b)$ of the vector initial value problem

$$y' = f(t, y)$$
$$y(a) = y_a$$

by the numerical solution vector u_n of Euler's method

$$u_{j+1} = u_j + hf(t_j, u_j) \qquad j = 0, 1, \ldots, n-1$$

with $h = (b-a)/n$ and $u_0 = y_a$.
(b) Use **euler.m** to approximate the solution $z(T)$ at $T = 4\pi$ of the initial value problem

$$z' = \begin{bmatrix} x \\ y \\ u \\ v \end{bmatrix}' = f(t, z) = \begin{bmatrix} u \\ v \\ -x/(x^2+y^2) \\ -y/(x^2+y^2) \end{bmatrix} \quad \text{with initial conditions } z = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

at $t = 0$ which cause the solution to move in a unit circle forever. Measure the maximum error

$$E_N = \max\left(x_N - \cos t_N, |y_n - \sin t_N|, |u_N + \sin t_N|, |v_N - \cos t_N|\right)$$

after 2 revolutions ($T = 4\pi$) with time steps $h = T/N$ for $N = 100, 200, \ldots, 800$. Estimate the constant C such that the error behaves like Ch. Measure the CPU time for each run and estimate the total CPU time necessary to obtain the solution to three-digit, six-digit, and twelve-digit accuracy. Plot the solutions.
(c) Use **euler.m** to verify conclusion (b) of problem 5.

Solution
The code for **euler.m** is on the following page. Following the code is a demonstration of the code and the solution it generates. The function used in the code following the **euler.m** function. The table for the time elapsed as a function of N, where N is the input size is also on the following page, with a plot as N ranges from $100, 200, \ldots, 800$. Using the The MATLAB program **euler.m** on the problem in 5b confirms our conclusion from the solution explodes when $t = \log(1 + 1/s^3)$. $\square$

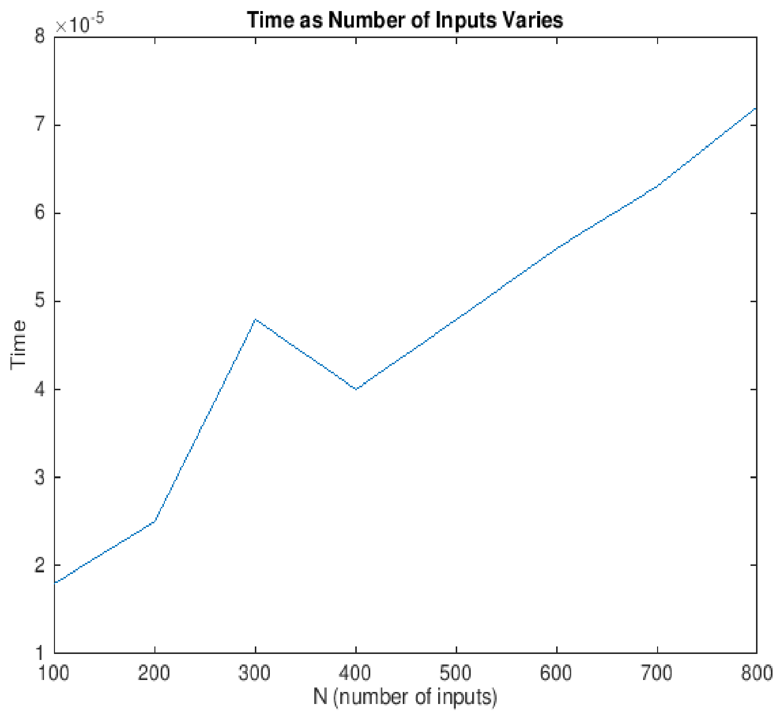
```
% approximate the solution of the ivp:
% y' = f(t,y) a <= t <= b, y(a) = alpha
function u = euler(a, b, ya, f, n)
    h = (b - a) / n;
    t = a;
    w_0 = ya;               % initial condition
    u = zeros(n+1, 2);

    u(1,1) = t;
    u(1,2) = w_0;

    for i = 2:n+1
        w_i = u(i-1,2) + h * f(u(i-1,2), t);
        t = t + h;
        u(i, 1) = t;         % update time
        u(i, 2) = w_i;       % update approximation
    end

    function y_p = f(y, t)
        y_p = y - t^2 + 1;
```

Table 1: Euler's Method of Approximation

Time	Euler	Actual
0	0.5000	0.5000
0.2	0.8000	0.8293
0.4	1.1520	1.2140
0.6	1.5504	1.6489
0.8	1.9885	2.1272
1.0	2.4582	2.6409
1.2	2.9498	3.1799
1.4	3.4518	3.7324
1.6	3.9501	4.2835
1.8	4.4282	4.8151
2.0	4.8658	5.3055

Table 2: Time as a function of input size

N	Time
100	0.000018
200	0.000025
300	0.000048
400	0.000040
500	0.000048
600	0.000056
700	0.000063
800	0.000072



Problem 8

The position $(x(t), y(t))$ of a satellite orbiting around the earth and moon is described by the second-order system of ordinary differential equations

$$x'' = x + 2y' - b\frac{x+a}{((x+a)^2+y^2)^{3/2}} - a\frac{x-b}{((x-b)^2+y^2)^{3/2}} \tag{9}$$

$$y'' = y - 2x' - b\frac{x+a}{((x+a)^2+y^2)^{3/2}} - a\frac{x-b}{((x-b)^2+y^2)^{3/2}} \tag{10}$$

where $a = 0.012277471$ and $b = 1 - a$. When the initial conditions $x(0) = 0.994, x'(0) = 0, y(0) = 0, y'(0) = -2.00158510637908$ are satisfied, there is a periodic orbit with period $T = 17.06521656015796$. Convert this problem to a 4×4 first-order system $u' = f(t, u), u(0) = u_0$ by introducing

$$u = [x, x', y, y'] = [u_1, u_2, u_3, u_4]$$

as a new vector unknown function and defining f appropriately.

Solution

Using equations (9) and (10) above, we can set up the matrix

$$\begin{bmatrix} x' \\ x'' \\ y' \\ y'' \end{bmatrix} = \begin{bmatrix} x' \\ x + 2y' - b\frac{x+a}{((x+a)^2+y^2)^{3/2}} - a\frac{x-b}{((x-b)^2+y^2)^{3/2}} \\ y' \\ y - 2x' - b\frac{x+a}{((x+a)^2+y^2)^{3/2}} - a\frac{x-b}{((x-b)^2+y^2)^{3/2}} \end{bmatrix}$$

The column vector on the left can be expressed as the derivative of the column vector u as defined above

$$u' = \begin{bmatrix} x \\ x' \\ y \\ y' \end{bmatrix}' = \begin{bmatrix} x' \\ x'' \\ y' \\ y'' \end{bmatrix} = \begin{bmatrix} x' \\ x + 2y' - b\frac{x+a}{((x+a)^2+y^2)^{3/2}} - a\frac{x-b}{((x-b)^2+y^2)^{3/2}} \\ y' \\ y - 2x' - b\frac{x+a}{((x+a)^2+y^2)^{3/2}} - a\frac{x-b}{((x-b)^2+y^2)^{3/2}} \end{bmatrix}$$

Replacing the values of the matrix u into the matrix on the right, we get the 4×4 first order system

$$u' = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}' = f(t, u) = \begin{bmatrix} u_2 \\ u_1 + 2u_4 - b\frac{u_1+a}{((u_1+a)^2+u_3^2)^{3/2}} - a\frac{u_1-b}{((u_1-b)^2+u_3^2)^{3/2}} \\ u_4 \\ y - 2u_2' - b\frac{u_1+a}{((u_1+a)^2+u_3^2)^{3/2}} - a\frac{u_1-b}{((u_1-b)^2+u_3^2)^{3/2}} \end{bmatrix}$$

$\square$