

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**  
**KHOA ĐIỆN TỬ - VIỄN THÔNG**  
**KHÓA NĂM HỌC 2023 – 2024**

-----o0o-----



**Đồ án môn học TH Phương pháp tính & MATLAB 2023**

**Đồ án 4:** Thiết kế giao diện triển khai **4 phương pháp giải gần đúng phương trình vi phân (O-le, hiện ẩn hình thang, hiện ẩn trung điểm, R-K)**. Giao diện gồm các **Edit Field (Text)** để nhập phương trình vi phân  $dy/dx = f(x,y)$ , điều kiện đầu  $y_0$ , giới hạn đầu cuối của mảng  $x$ , số đoạn con  $N$ , sai số (dùng cho pp hiện ẩn hình thang). Sử dụng **List Box** để lựa chọn 1 trong 4 phương pháp. **Sử dụng duy nhất 1 Button** để:

- ❖ Giải phương trình theo phương pháp đã chọn.
- ❖ Vẽ đồ thị gần đúng  $y = f(x)$

Giảng viên : Th.s Thái Hồng Hải  
Sinh viên thực hiện : Nguyễn Tiến Đại  
MSSV : 21200274

**Thành phố Hồ Chí Minh, năm 2023**

## 1. Code thuật toán

### a. Phương pháp Ô-le

\* Code:

```
function [x,y] = Phuong_Phap_Ole(fxy, x0, xn, y0, N)
    h = (xn - x0)/N;
    x = x0:h:xn;
    y = zeros(1, length(x));
    y(1) = y0;
    for i = 1:N
        y(i+1) = y(i) + h * fxy(x(i), y(i));
    end
end
```

\* Giải thích:

- Khai báo hàm với 5 thông số vào: fxy, x0, xn, y0 và N
- + fxy: Hàm  $f(x,y)$  trong phương trình vi phân  $y' = fxy$
- + x0 và xn: Giá trị đầu và cuối của mảng x
- + y0: Giá trị điều kiện ban đầu
- + N: Số đoạn con được chia
- Đầu ra có x, y
- Tính bước nhảy h dựa vào x0, xn, N:  $h = \frac{x_n - x_0}{N}$
- Khởi tạo mảng x chứa các giá trị từ x0 đến xn bước nhảy h
- Khởi tạo mảng y với kích thước bằng độ dài mảng x và các phần tử ban đầu bằng 0
- Gán giá trị ban đầu y0 cho phần tử đầu tiên của mảng y
- Cho vòng lặp for từ 1:N. Trong mỗi vòng lặp, giá trị gần đúng tại vị trí i+1 được tính bằng công thức Ô-le:  $y(i+1) = y(i) + h * fxy(x(i), y(i))$

### b. Phương pháp hiện ẩn trung điểm

\* Code:

```
function [x,y] = Hien_An_Trung_Diem(fxy, x0, xn, y0, N)
    h = (xn - x0)/N;
    x = x0:h:xn;
    y = zeros(1, length(x));
    y(1) = y0;
    for i = 1:N
        z(i + 1) = y(i) + (h/2) * fxy(x(i), y(i));
        y(i + 1) = y(i) + h * fxy(x(i) + (h/2), z(i + 1));
    end
end
```

end

\* Giải thích:

- Các thông số vào và ra cũng như phương pháp O-le
- Cho vòng lặp for từ 1 đến N:
  - + Giá trị trung điểm  $z(i+1)$  dùng để ước lượng của  $y$  tại giữa 2 điểm  $x(i)$  và  $x(i+1)$  và được tính dựa vào giá trị  $y(i)$  và hàm  $f_{xy}(x(i), y(i))$ :  $z(i+1) = y(i) + \frac{h}{2} f_{xy}(x(i), y(i))$
  - + Giá trị gần đúng của  $y$  tại  $i+1$  bằng công thức hiện ẩn trung điểm:

$$y(i+1) = y(i) + h * f_{xy}(x(i) + \frac{h}{2}, z(i+1))$$

Giá trị mới của  $y$  dựa vào  $f_{xy}$  và giá trị trước đó  $z(i+1)$  được đánh giá tại trung điểm  $x(i) + \frac{h}{2}$  và  $z(i+1)$

c. Phương pháp hiện ẩn hình thang

\* Code:

```
function [x, y] = Hien_An_Hinh_Thang(fxy, x0, xn, y0, N, e)
h = (xn - x0)/N;
x = x0:h:xn;
y = zeros(1, length(x));
y(1) = y0;
for i = 1:N
    z(i+1) = y(i) + h * fxy(x(i), y(i));
    y(i+1) = y(i) + (h/2) * (fxy(x(i), y(i)) + fxy(x(i+1),
z(i+1)));
    if abs(y(i+1) - z(i+1)) <= e
        break
    end
end
end
```

\* Giải thích:

- Các thông số vào và ra cũng như phương pháp O-le nhưng thêm thông số đầu vào là sai số  $e$
- Cho vòng lặp for từ 1 đến N:
  - + Giá trị  $z(i+1)$  là ước lượng của  $y$  tại điểm  $x(i+1)$  sử dụng phương pháp O-le

$$z(i+1) = y(i) + h * f_{xy}(x(i), y(i))$$

- + Giá trị mới của  $y$  dựa vào giá trị trước đó, hàm  $f_{xy}$  được đánh giá tại 2 điểm  $x(i)$ ,  $x(i+1)$  và giá trị  $z(i+1)$  trên. Công thức hiện ẩn hình thang:

$$y(i+1) = y(i) + \frac{h}{2} * [f_{xy}(x(i), y(i)) + f_{xy}(x(i+1), z(i+1))]$$

+ Kiểm tra độ chính xác bé hơn sai số  $\epsilon$ , vòng lặp sẽ dừng lại bằng lệnh break

#### d. Phương pháp Runge-Kutta (R-K)

\* Code:

```
function [x,y] = Phuong_Phap_RK(fxy, x0, xn, y0, N)
h = (xn - x0)/N;
x = x0:h:xn;
y = zeros(1, length(x));
y(1) = y0;
    for i = 1:N
        k1 = h * fxy(x(i), y(i));
        k2 = h * fxy(x(i) + h/2, y(i) + k1/2);
        k3 = h * fxy(x(i) + h/2, y(i) + k2/2);
        k4 = h * fxy(x(i) + h, y(i) + k3);
        y(i + 1) = y(i) + (k1 + 2*k2 + 2*k3 + k4)/6;
    end
end
```

\* Giải thích:

- Các thông số vào và ra cũng như phương pháp O-le

- Cho vòng lặp for từ 1 đến N:

+ Giá trị  $k_1$  phụ thuộc vào  $y(i)$  và hàm  $f_{xy}(x(i), y(i))$

**$k_1 = h * f_{xy}(x(i), y(i))$  : độ dốc của hàm  $y$  tại  $x(i)$**

+  $k_2$  dựa vào  $y(i)$ ,  $k_1$  tính trên và hàm  $f_{xy}$

**$k_2 = h * f_{xy}(x(i) + \frac{h}{2}, y(i) + \frac{k_1}{2})$  : độ dốc của hàm  $y$  tại điểm trung bình  $x(i)$  và  $x(i+1)$**

+  $k_3$  sử dụng  $y(i)$ ,  $k_2$  và hàm  $f_{xy}$  tại điểm trung bình giữa  $x(i)$  và  $x(i+1)$

**$k_3 = h * f_{xy}(x(i) + \frac{h}{2}, y(i) + \frac{k_2}{2})$**

+  $k_4$  dựa trên giá trị  $y(i)$ ,  $k_3$  và  $f_{xy}$  tại điểm  $x(i+1)$

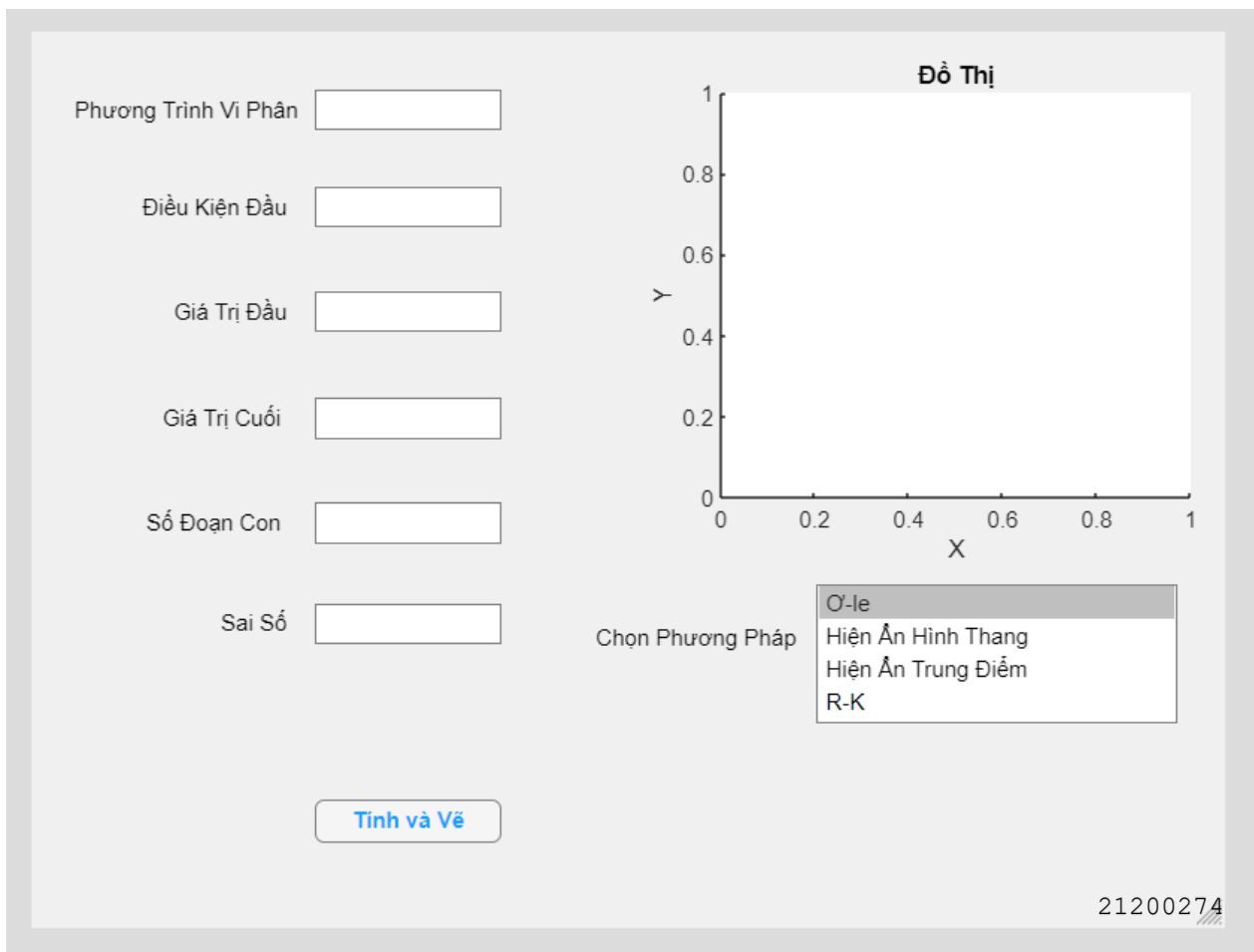
**$k_4 = h * f_{xy}(x(i) + h, y(i) + k_3)$**

+ Giá trị gần đúng của  $y$  tại vị trí  $i+1$  bằng công thức trung bình của các ước lượng độ dốc

**$y(i+1) = y(i) + \frac{1}{6} (k_1 + 2k_2 + 2k_3 + k_4)$**

## 2. Code giao diện

a. Giao diện app



- Tạo đồ thị  $f(x)$  với nhãn `app.DoThiUIAxes`
- Tạo các thông số đầu vào  $fxy$ ,  $y_0$ ,  $x_0$ ,  $x_n$ ,  $N$ ,  $e$  với nhãn lần lượt là `app.PhuongTrinhViPhanEditField`, `app.DieuKienDauEditField`, `app.GiaTriDauEditField`, `app.GiaTriCuoiEditField`, `app.SoDoanConEditField`, `app.SaiSoEditField`
- Tạo nút nhấn tính kết quả và vẽ với nhãn `app.TinhvaVeButton`
- Để chọn lựa phương pháp dùng List Box tên nhãn `app.ChonPhuongPhapListBox`

b. Nội dung code:

\* Code:

```
function TinhvaVeButtonPushed(app, event)
    f = app.PhuongTrinhViPhanEditField.Value;
    fxy = str2func(['@(x,y)', f]);
    y0 = str2double(app.DieuKienDauEditField.Value);
    x0 = str2double(app.GiaTriDauEditField.Value);
    xn = str2double(app.GiaTriCuoiEditField.Value);
    N = str2double(app.SoDoanConEditField.Value);
    e = str2double(app.SaiSoEditField.Value);
```

```

temp = app.ChonPhuongPhapListBox.Value;
if (temp == "Ô-le")
    [x,y] = Phuong_Phap_Ole(fxy, x0, xn, y0, N);
    plot(app.DoThiUIAxes, x, y, 'k*--');
elseif (temp == "Hiện Ảnh Hình Thang")
    [x,y] = Hien_An_Hinh_Thang(fxy, x0, xn, y0, N, e);
    plot(app.DoThiUIAxes, x, y, 'ro--');
elseif (temp == "Hiện Ảnh Trung Điểm")
    [x,y] = Hien_An_Trung_Diem(fxy, x0, xn, y0, N);
    plot(app.DoThiUIAxes, x, y, 'b+--');
elseif (temp == "R-K")
    [x,y] = Phuong_Phap_RK(fxy, x0, xn, y0, N);
    plot(app.DoThiUIAxes, x, y, 'gv--');
end
end
end

```

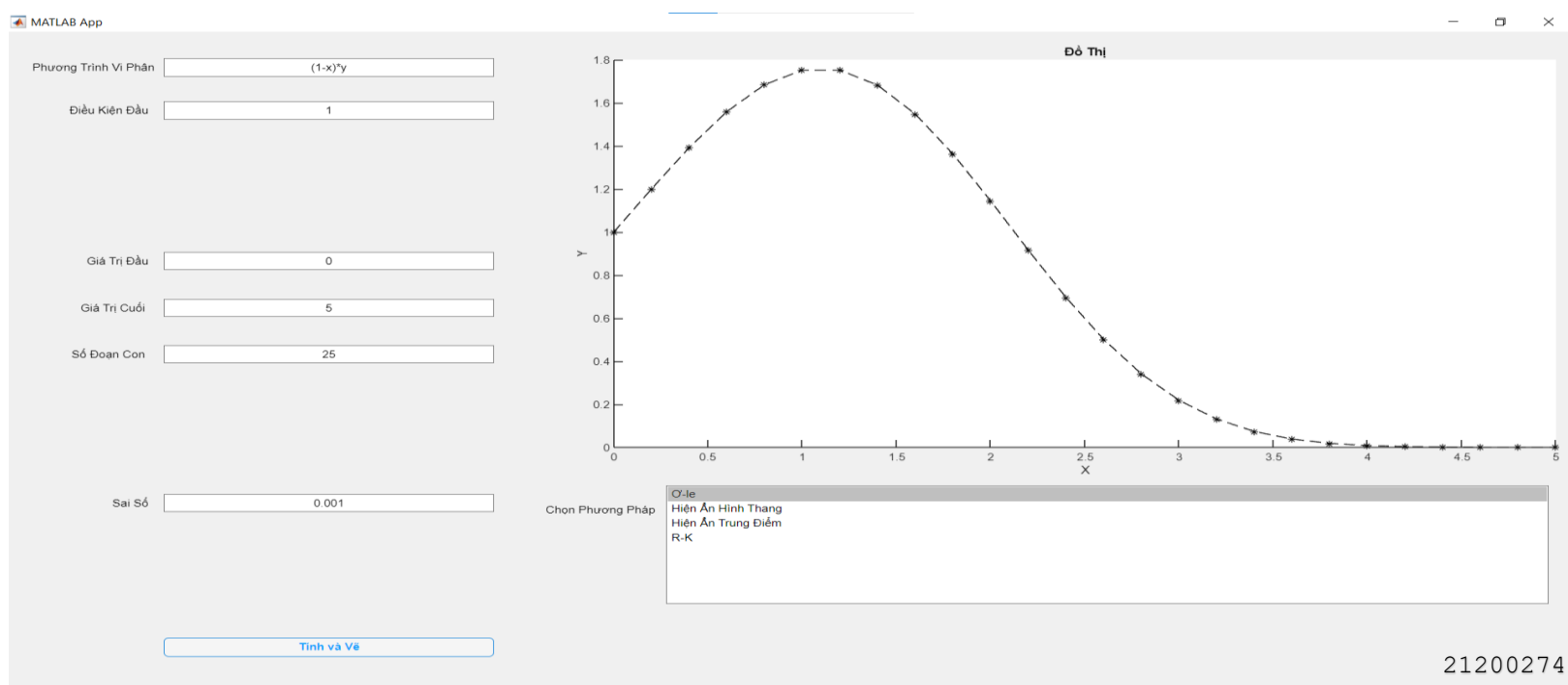
\* Giải thích:

- Gán giá trị f bằng app.PhuongTrinhViPhanEditField.Value và đổi kiểu dữ liệu từ string sang functionhandle.
- Gán các giá trị y0, x0, xn, N, e lần lượt bằng app.DieuKienDauEditFiled.Value, app.GiaTriDauEditFiled.Value, app.GiaTriCuoiEditFiled.Value, app.SoDoanConEditFiled.Value, app.SaiSoEditFiled.Value ở dạng chuỗi sang số thập phân với độ chính xác kép.
- Gán giá trị temp bằng app.ChonPhuongPhapListBox.Value và kiểm tra giá trị của temp
- Nếu temp có giá trị “Ô-le” thì hàm Phuong\_Phap\_Ole sẽ được gọi đến với các thông số đầu vào là fxy, x0, xn, y0, N để tính toán các giá trị x, y. Sau đó vẽ x, y lên đồ thị bằng câu lệnh Plot

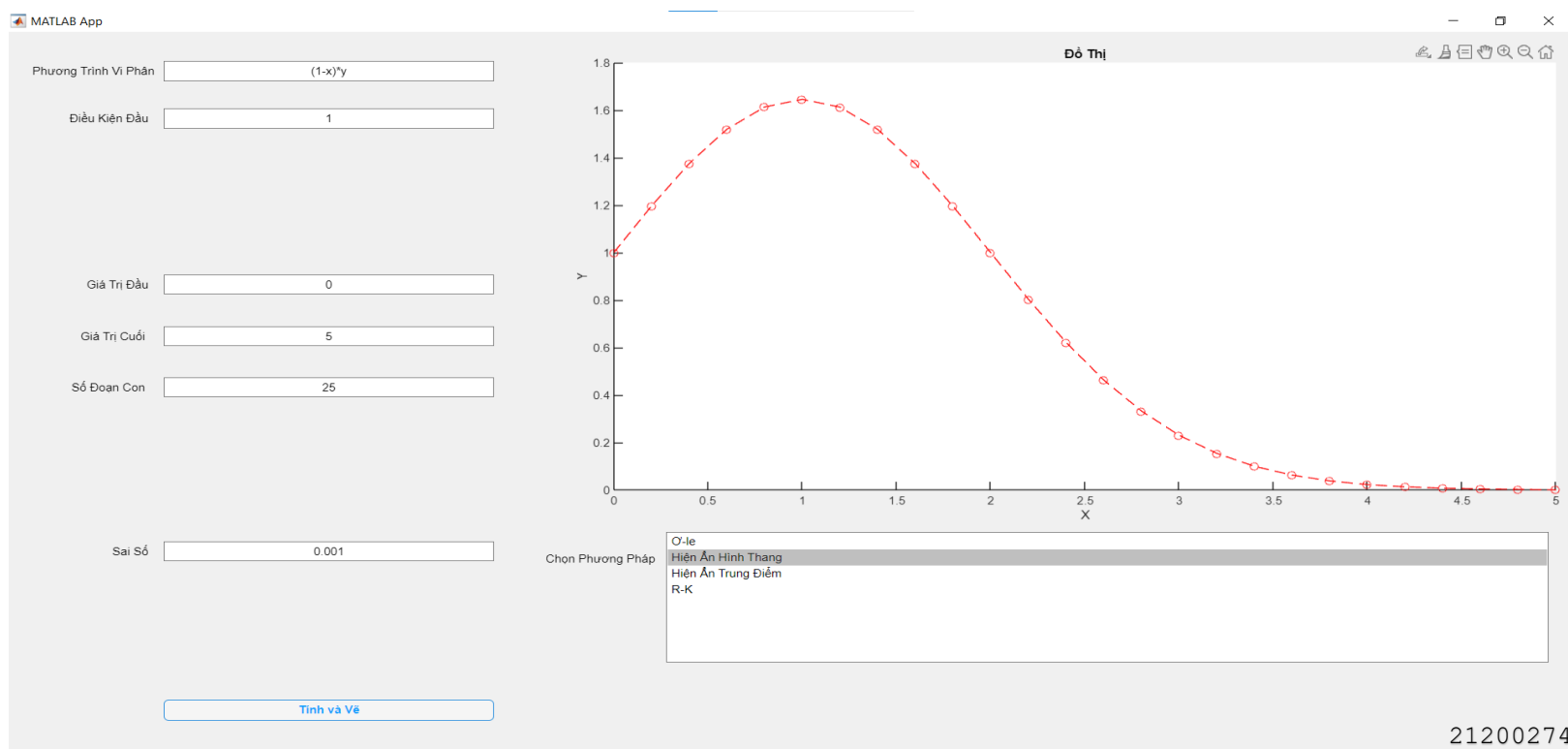
c. Kết quả mô phỏng:

Với  $fxy = (1-x)*y$        $y(0) = 1$        $x0 = 0$        $xn = 5$        $N = 25$        $e = 0.001$

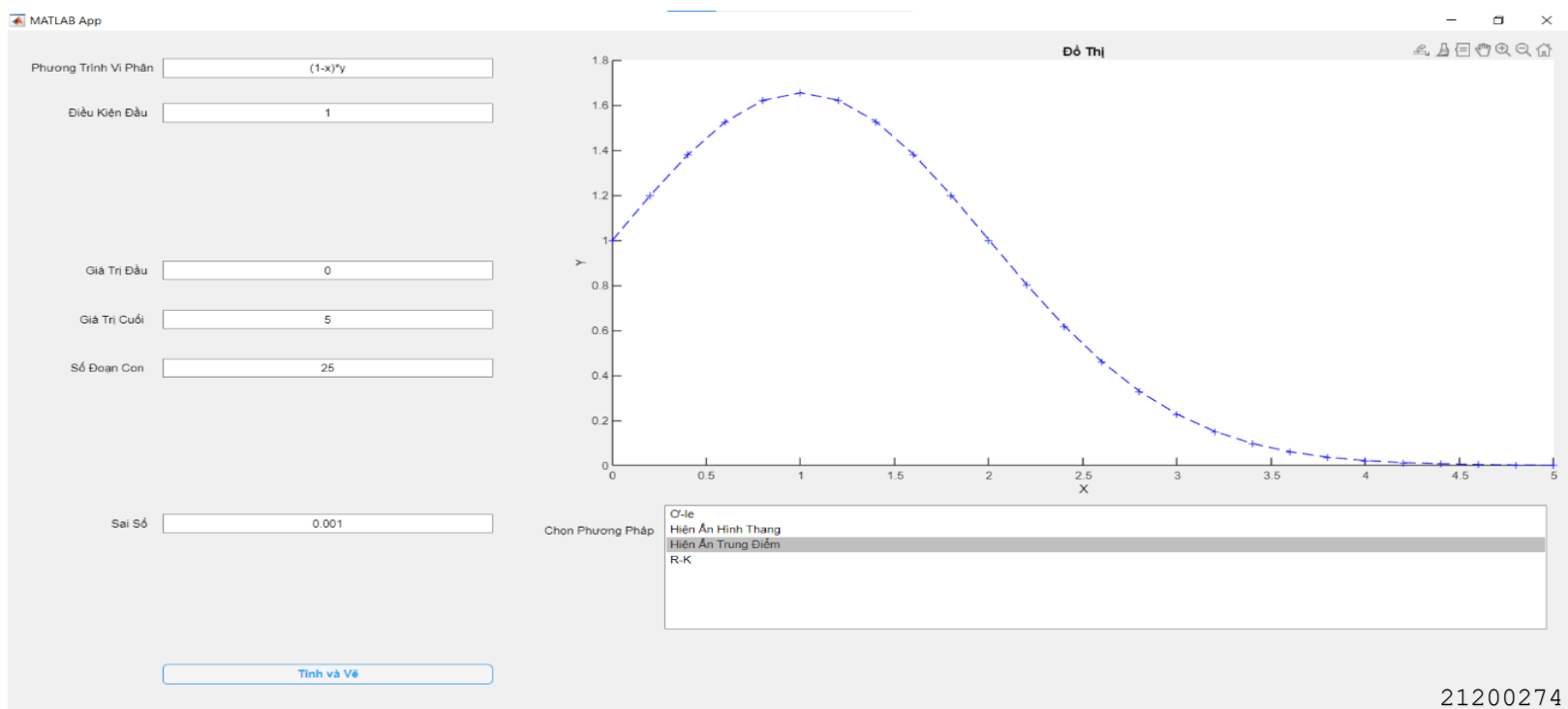
- Phương pháp Ô-le:



- Phương pháp hiện ẩn hình thang:

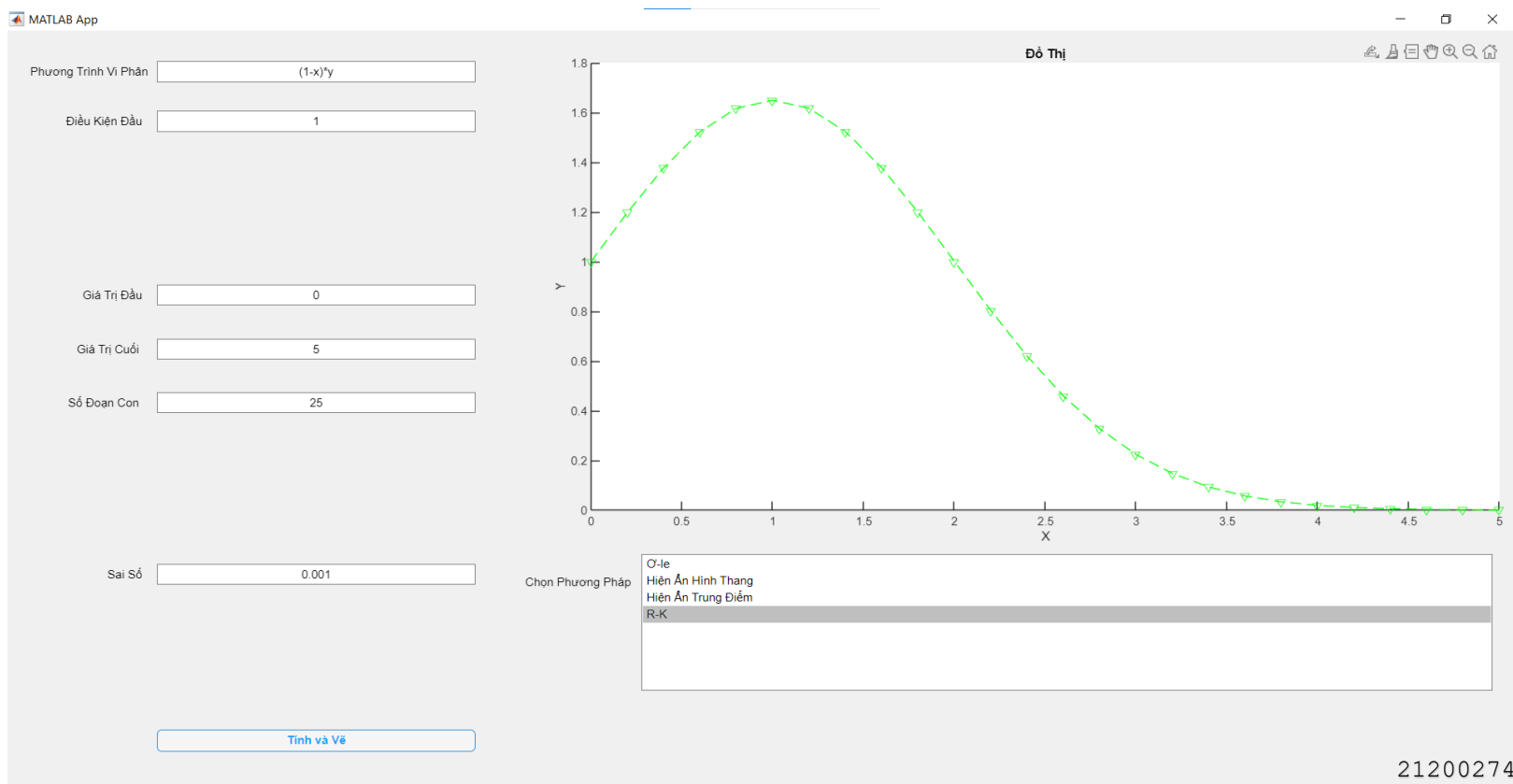


- Phương pháp hiện ẩn trung điểm:



21200274

## - Phương pháp R-K:



21200274