Elliptic Grid Generation

The process of grid generation that we study can be described by the following steps:

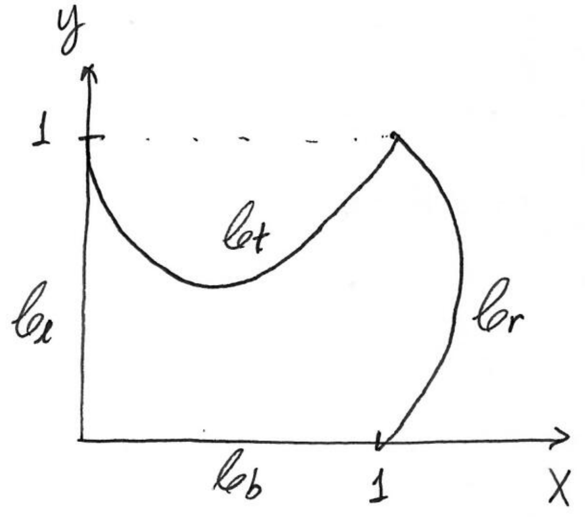
I) Describe the boundary ∂D of the physical domain D, by parametric equations:

$$\partial D: \ \vec{r}(t) = (x(t), y(t)), \quad t \in [a,b].$$

Depending on the boundary more than one parametric equation will be needed.

For the simply connected domain, we will have four pieces defining the boundary

For example, the "Swan" domain



$l_l: \ x_l(t) = 0, \ y_l(t) = t, \ 0 \le t \le 1$

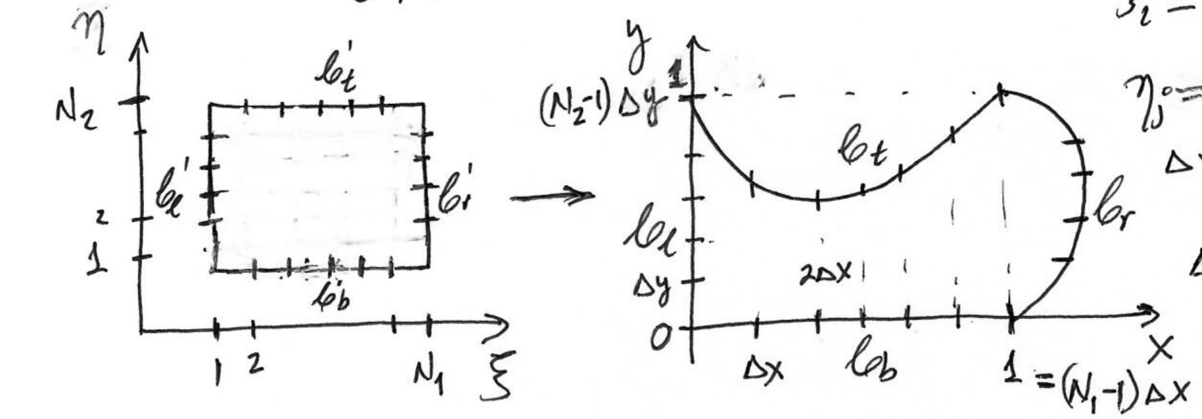
$l_b: \ x_b(t) = t, \ y_b(t) = 0, \ 0 \le t \le 1$

$l_r: \ x_r(t) = 1 + 2t - 2t^2, \ y_r(t) = t, \ 0 \le t \le 1$

$l_t: \ x_t(t) = t, \ y_t(t) = 1 - 2t + 2t^2, \ 0 \le t \le 1$

II) Define a transformation ∂T from the boundary of the rectangular region $\partial D'$ to the boundary of the physical domain ∂D.

$$\partial T: \partial D' \longrightarrow \partial D$$



$\xi_i = i, \quad 1 \le i \le N_1$

$\eta_j = j, \quad 1 \le j \le N_2$

$\Delta x = \dfrac{1}{N_1 - 1}$

$\Delta y = \dfrac{1}{N_2 - 1}$

$$\partial T: \quad \partial D' \longrightarrow \partial D$$

b_b: $(\xi_i, 1) \longrightarrow (x(\xi_i,1),\ y(\xi_i,1)) = ((i-1)\Delta x, 0)$

$(i,1) \longrightarrow (x(i,1),\ y(i,1)) =$

(3.1) b_r: $(\xi_{N_1}, \eta_j) \longrightarrow (x(\xi_{N_1},\eta_j),\ y(\xi_{N_1},\eta_j))$

$(N_1, j) \longrightarrow (x(N_1,j),\ y(N_1,j)) =$

$(1 + 2y(N_1,j) - 2y^2(N_1,j),\ (j-1)\Delta y)$

b_t: $(i, N_2) \longrightarrow (x(i,N_2),\ y(i,N_2)) =$

$((i-1)\Delta x,\ 1 - 2x(i,N_2) + 2x^2(i,N_2))$

b_l: $(1, j) \longrightarrow (x(1,j),\ y(1,j)) = (0, (j-1)\Delta y)$

III) The transformation ∂T is extended to a transformation T including all the interior points of D' and D.

$$T: D' \longrightarrow D$$
$$(\xi,\eta) \longrightarrow (x(\xi,\eta), y(\xi,\eta))$$

In our case, $x(\xi,\eta)$ and $y(\xi,\eta)$ are defined from the solution of a BVP of a system of elliptic equations:

a) Amsden - Hirt: Two Laplace equations

$$\nabla^2_{\xi\eta} x = x_{\xi\xi} + x_{\eta\eta} = 0 \qquad (4.1)$$
$$\nabla^2_{\xi\eta} y = y_{\xi\xi} + y_{\eta\eta} = 0$$

b) Winslow: Two quasi-linear elliptic equations.

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = 0$$
$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = 0$$

where $\alpha = x_\eta^2 + y_\eta^2, \quad \beta = x_\xi x_\eta + y_\xi y_\eta$
$\gamma = x_\xi^2 + y_\xi^2 .$

Amsden - Hirt Grid Generator

the BVP consists of the system of equations (4.1) and the continuous version of (3.1).

Why elliptic?

a) Maximum Principle.

b) Smoothness of coordinates obtained as solutions of BVP.

A disadvantage is that there is no control on the grid points location in the interior of D.

The BVP (4.1) with continuous version of B.C. (3.1) is solved by centered 2nd order finite difference

For example discretization of x using 5-point stencil

$$\frac{1}{\Delta\xi^2}\left(x_{i-1,j} - 2x_{i,j} + x_{i+1,j}\right) + \frac{1}{\Delta\eta^2}\left(x_{i,j-1} - 2x_{ij} + x_{i,j+1}\right) = 0$$

$$2 \le i \le N_1 - 1, \quad 2 \le j \le N_2 - 1 \qquad (5.1)$$

There is a similar equation for the y-coordinate.

Equation (5.1) reduces to

$$\Delta\eta^2 \left(x_{i-1,j} - 2x_{ij} + x_{i+1,j} \right) + \Delta\xi^2 \left(x_{i,j-1} - 2x_{ij} + x_{i,j+1} \right) = 0 \tag{6.1}$$

This discrete equation approximating Laplace equation and another completely analogous for the coordinate y together with boundary conds. (3.1) can be solved direct methods for sparse linear systems such as those studied previously.

However, we will apply iterative methods (SOR) to obtain their solutions.

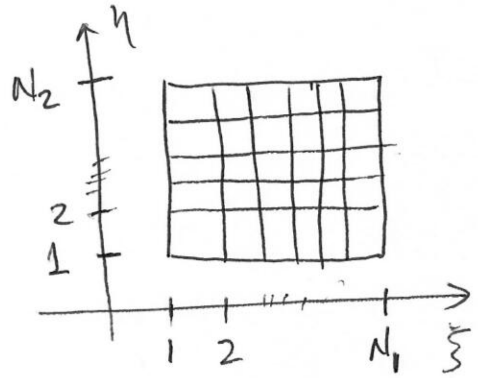
Thus, solving for x_{ij} from (6.1)

$$-2\left(\Delta\eta^2 + \Delta\xi^2\right) x_{ij} = -\Delta\eta^2 \left(x_{i-1,j} + x_{i+1,j} \right) - \Delta\xi^2 \left(x_{i,j-1} + x_{i,j+1} \right)$$

or

$$x_{ij} = \frac{\Delta\eta^2}{2(\Delta\eta^2 + \Delta\xi^2)} \left(x_{i-1,j} + x_{i+1,j} \right) + \frac{\Delta\xi^2}{2(\Delta\eta^2 + \Delta\xi^2)} \left(x_{i,j-1} + x_{i,j+1} \right) \tag{6.2}$$

In all our experiments, we will use as our Computational domain a rectangular region in the $\xi\eta$-plane such as

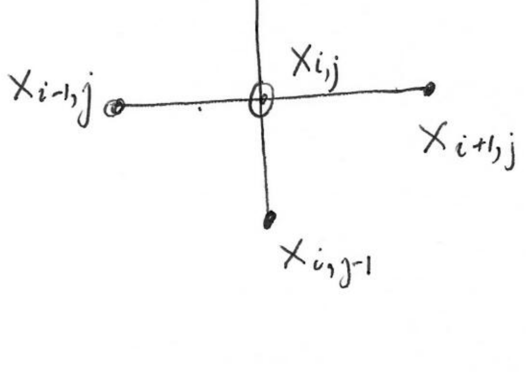


Thus, $\Delta\xi = 1$, $\Delta\eta = 1$.

Therefore, (6.2) reduces to

$$X_{ij} = \frac{X_{i-1,j} + X_{i+1,j} + X_{i,j-1} + X_{i,j+1}}{4} \quad (7.1)$$

It means X_{ij} is the average of its four neighboors points.



We have already discussed (previous sections) the application of iterative methods to (7.1).
In fact, we found that for the pentadiagonal matrix corresponding to (5.1) SOR method converges for any initial guess and there is an optimum ω given by

$$\omega_{opt} = \frac{4}{2+\sqrt{4-\left(\cos\frac{\pi}{N_1}+\cos\frac{\pi}{N_2}\right)^2}}$$

Applying SOR iteration:

$$\hat{x}_{ij} = \frac{1}{4}\left(x_{i,j+1}^{(k-1)} + x_{i,j-1}^{(k)} + x_{i+1,j}^{(k-1)} + x_{i-1,j}^{(k)}\right)$$

$$\hat{y}_{ij} = \frac{1}{4}\left(y_{i,j+1}^{(k-1)} + y_{i,j-1}^{(k)} + y_{i+1,j}^{(k-1)} + y_{i-1,j}^{(k)}\right)$$

$$x_{ij}^{(k)} = \omega\,\hat{x}_{ij} + (1-\omega)\,x_{i,j}^{(k-1)}.$$

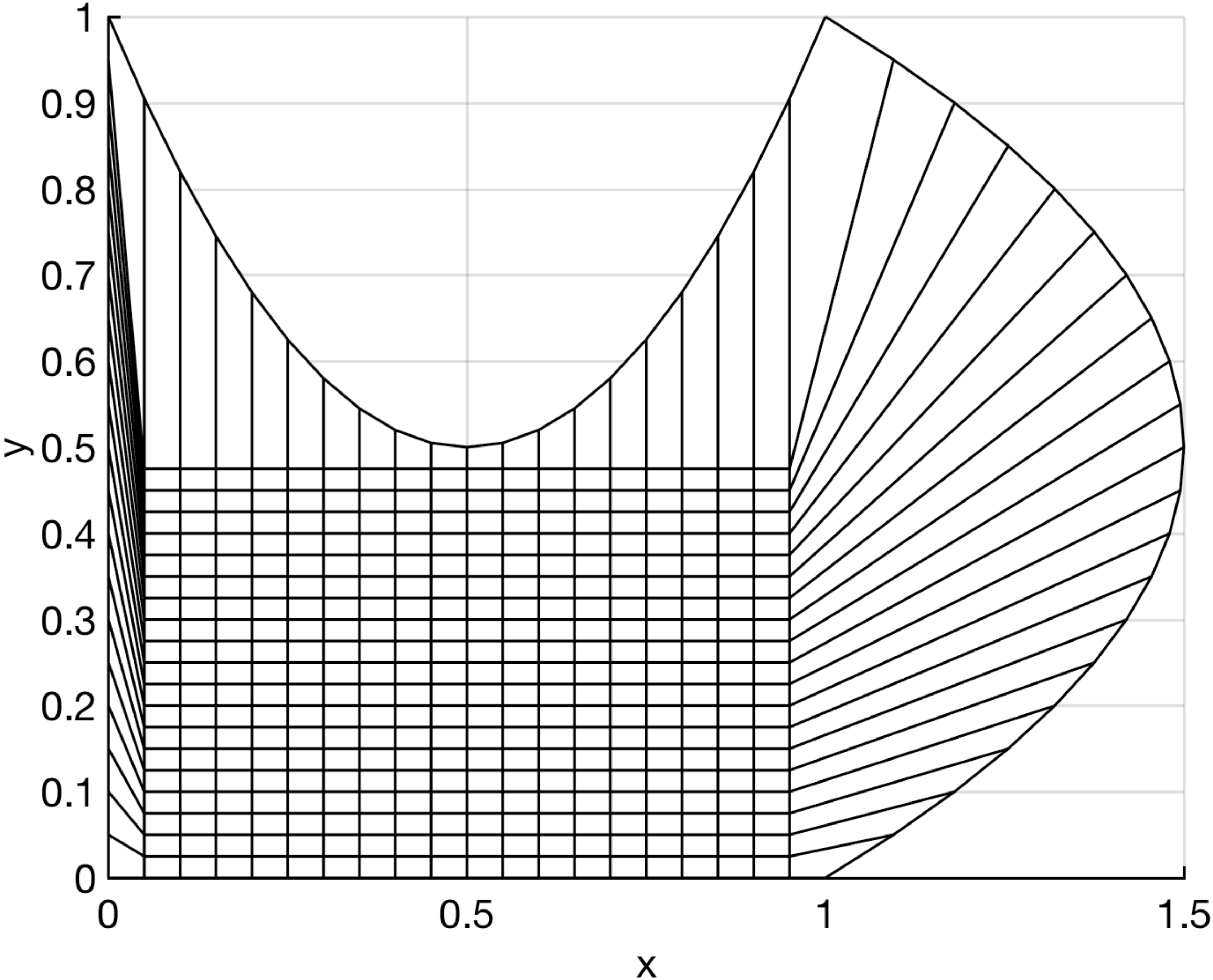
$$y_{ij} = \omega\,\hat{y}_{ij} + (1-\omega)\,y_{i,j}^{(k-1)}$$

$$2 \le i \le N_1-1, \qquad 2 \le j \le N_2-1., \qquad 0<\omega<2$$

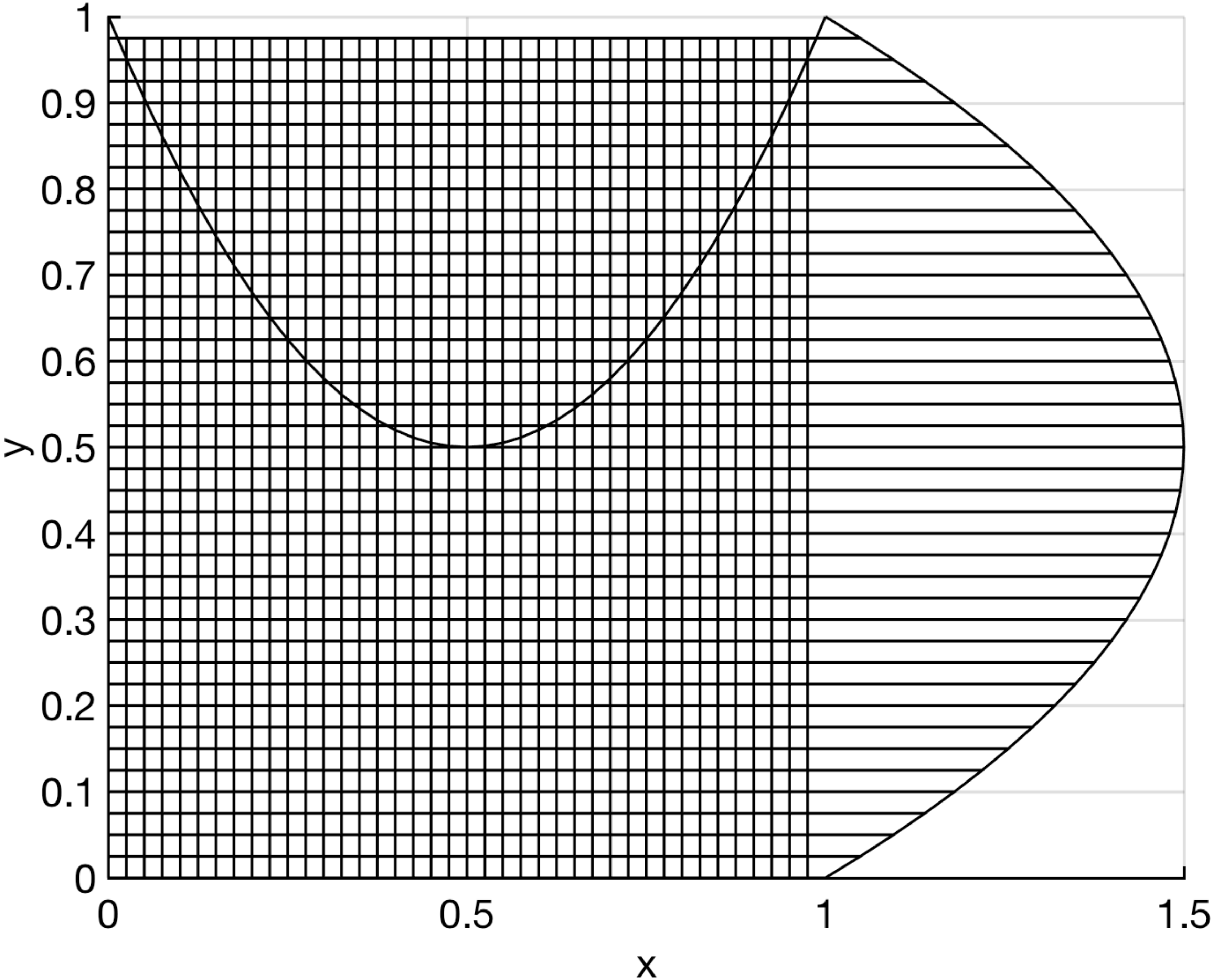
IV) To start this SOR iterative method, we need an initial grid. According to the theoretical results on convergence, this initial grid can be arbitrary as long as $0<w<2$.

I am including two of these grids that you can use for your particular code.

V) As a final step on your algorithm, you need to stop the process when convergence has been reached. In the next page, I am including a stop criteria.



y
x
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.5
1
1.5



1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
y
0
0.5
1
1.5
x

STOP CRITERIA

More precisely, uniform convergence is established when two consecutive grids agree within an specified tolerance

$$\max_{\substack{2 \le i \le N_1 - 1 \\ 2 \le j \le N_2 - 1}} \left| x_{i,j}^{(k)} - x_{i,j}^{(k-1)} \right| < TOL$$

and

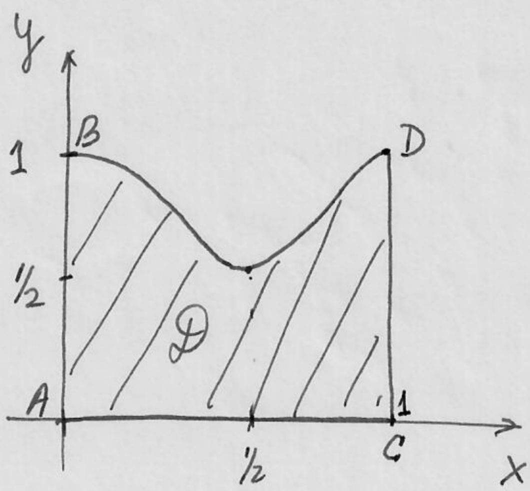
$$\max_{\substack{2 \le i \le N_1 - 1 \\ 2 \le j \le N_2 - 1}} \left| y_{ij}^{(k)} - y_{ij}^{(k-1)} \right| < TOL$$

$k = 1, 2, \ldots$

Some interesting domains where boundary conforming coordinates will be generated.

A) Simply connected.

A1) Valley domain



Top Boundary: (BD).

$x_t(s) = s$

$y_t(s) = \frac{3}{4} + \frac{1}{4} \sin\left(\pi\left(\frac{1}{2} + 2s\right)\right)$

$0 \le s \le 1$

Bottom boundary:

$x_b(s) = s, \quad y_b(s) = 0, \quad 0 \leq s \leq 1$

Left boundary:

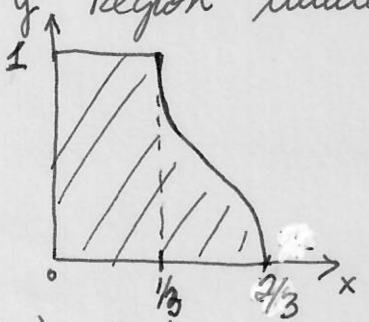
$x_\ell(s) = 0, \quad y_\ell(s) = s, \quad 0 \leq s \leq 1$

Right boundary:

$x_r(s) = 1, \quad y_r(s) = s, \quad 0 \leq s \leq 1$

A2) ALMOST Convex region:

Region limited by i) the coordinate axes $\begin{cases} x = 0 \\ y = 0 \end{cases}$



ii) line $y = 1$

iii) Curve: $x = \frac{1}{2} + \frac{1}{6}\cos(\pi y)$.

A3) Antenna Domain:

Region limited by the segments:

i) $y = 1, \quad -3 \leq x \leq 0$

ii) $y = 0, \quad -3 \leq x \leq 0.$

iv) $x = 0$

$1 \leq y \leq 3 + 1/2$

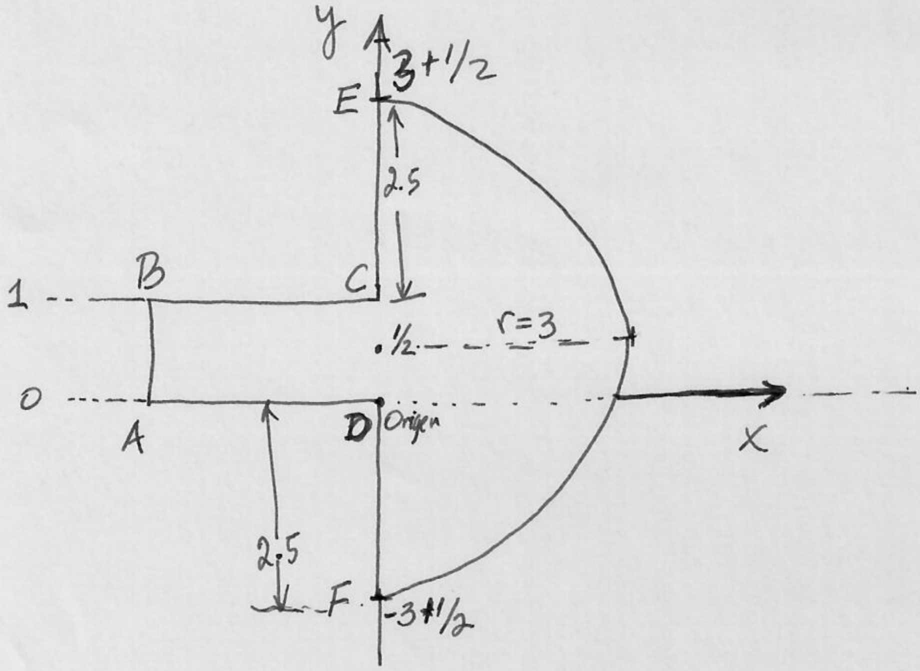
v) $x = 0$

$-3 + \frac{1}{2} \leq y \leq 0.$

vi) Semicircle

$x \geq 0$

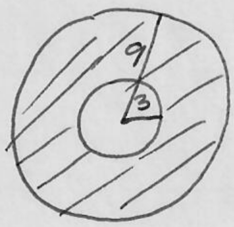
$x^2 + (y - 1/2)^2 = 3$



B) Multiply Connected domains:

B1) Annular region between two concentric Circles:

$$x^2+y^2 \geq 3^2, \qquad x^2+y^2 \leq 9^2$$



B2) Region between three-leafed rose and outer circle

Three-leafed Rose.
$$\begin{cases} x(t) = 0.3\,(2+\cos(3t))\cos t, \\ y(t) = 0.3\,(2+\cos(3t))\sin t., \end{cases} \qquad 0 \leq t \leq 2\pi$$

Outer circle:

$$x^2+y^2 = 9^2 \text{ or}$$

$$x(t) = 9\cos t, \qquad 0 \leq t \leq 2\pi.$$
$$y(t) = 9\cos t,$$

A4) Swan Domain

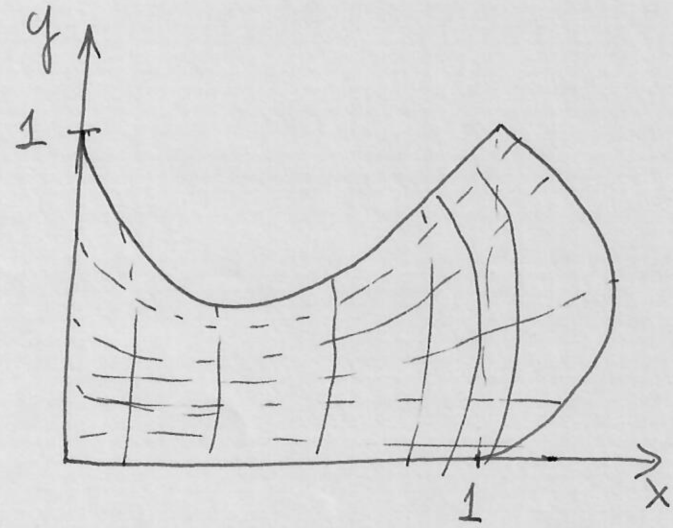
Region limited by the curves:

i) $y = 0, \quad 0 \le x \le 1$ (Bottom boundary)

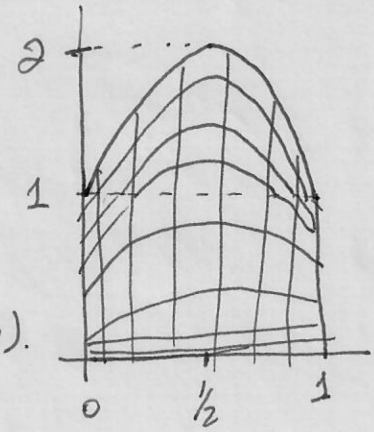
ii) $y = 1 - 2x + 2x^2, \quad 0 \le x \le 1.$ (Top boundary)

iii) $x = 0, \quad 0 \le y \le 1,$ (Left boundary)

iv) $x = 1 + 2y - 2y^2, \quad 0 \le y \le 1$ (Right boundary).



A5) Dome (convex region):



i) $y = 0, \quad 0 \le x \le 1$ (Bottom bdry).

ii) $y = -4(x - 1/2)^2 + 2, \quad 0 \le x \le 1$ (Top bdry).

iii) $x = 0, \quad 0 \le y \le 1,$ (left bdry).

iv) $x = 1, \quad 0 \le y \le 1,$ (right bdry)

Since the technique previously described (Iterative) requires an initial guess grid. This needs to be provided as an initial data.

Show various initial grids from MATLAB Simulations.

You will notice that for most of these domains we obtain Convergence to a final grid regardless of the initial grid.

Two possible problems:

i) A point inside the logical space is mapped to a point outside of the physical domain.

ii) Two or more point in the computational domain are mapped to the same point in the physical domain. These are called folded grids.

It is desirable to have a mapping that is

a) One-to-one.

b) onto. Any point of physical domain is an image of a point in the comp. domain.