

Eine Testaufgabe zur FEM in zwei Dimensionen

Aufgabe 1

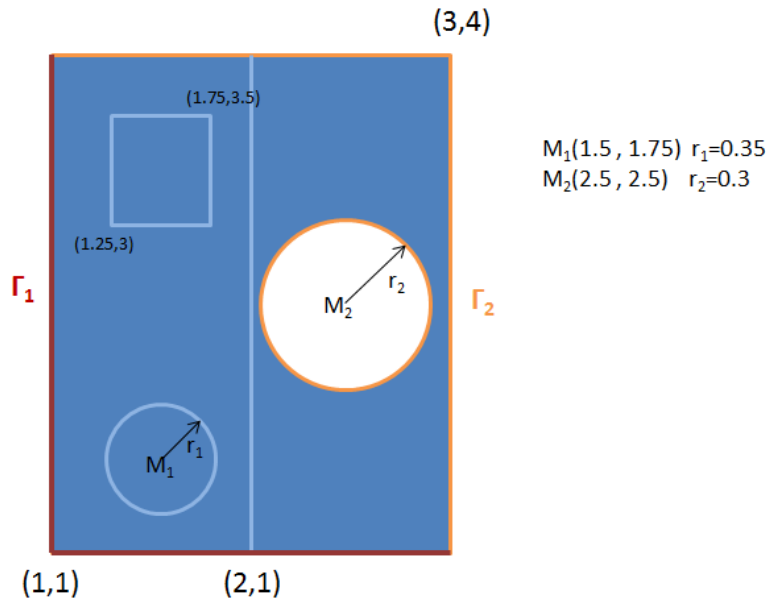
Gegeben sei folgende partielle Differentialgleichung:

$$-\frac{\partial}{\partial x} \left(\alpha_1(x, y) \frac{\partial \Phi}{\partial x} \right) - \frac{\partial}{\partial y} \left(y^2 \frac{\partial \Phi}{\partial y} \right) + \beta(x, y) \Phi = f(x, y)$$

Für die Funktionen $\alpha_1(x, y)$, $\beta(x, y)$, $f(x, y)$ gelte folgendes:

$$\alpha_1(x, y) = \begin{cases} 0.01 & 1.25 \leq x \leq 1.75 \text{ und } 3 \leq y \leq 3.5 \\ 5yx^2 & \text{sonst} \end{cases} \quad \beta(x, y) = \begin{cases} 500 & (x-1.5)^2 + (y-1.75)^2 \leq 0.35^2 \\ 5 & \text{sonst} \end{cases}$$

$$f(x, y) = \begin{cases} -10xy & x \geq 2 \\ 0 & \text{sonst} \end{cases}$$



a) Lösen Sie die obige Differentialgleichung im gezeigten Gebiet mit der Randbedingung $\Phi|_{\Gamma_1 \cup \Gamma_2} = x^2 - y^2 + 1$

b) Lösen Sie die obige Differentialgleichung im gezeigten Gebiet mit der Randbedingung $\Phi|_{\Gamma_2} = x^2 - y^2 + 1$ und $\left(\alpha_1 \frac{\partial \Phi}{\partial x}, \alpha_2 \frac{\partial \Phi}{\partial y} \right) \cdot \vec{n} + 3xy\Phi|_{\Gamma_1} = 20$

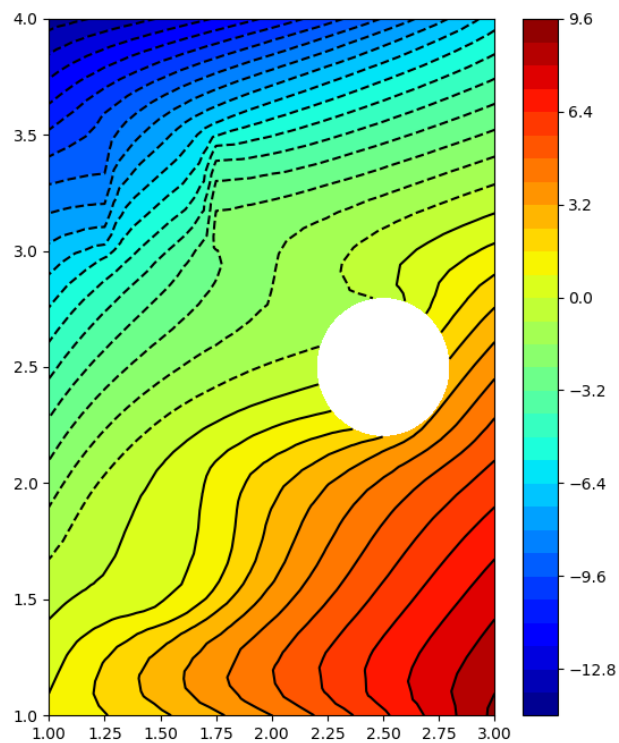
c) Erstellen Sie mit meshtools ein Netz des obigen Gebietes und geben Sie die zu Γ_1 bzw. Γ_2 gehörigen Randsegmente/knoten aus.

Hinweis zu den Lösungen:

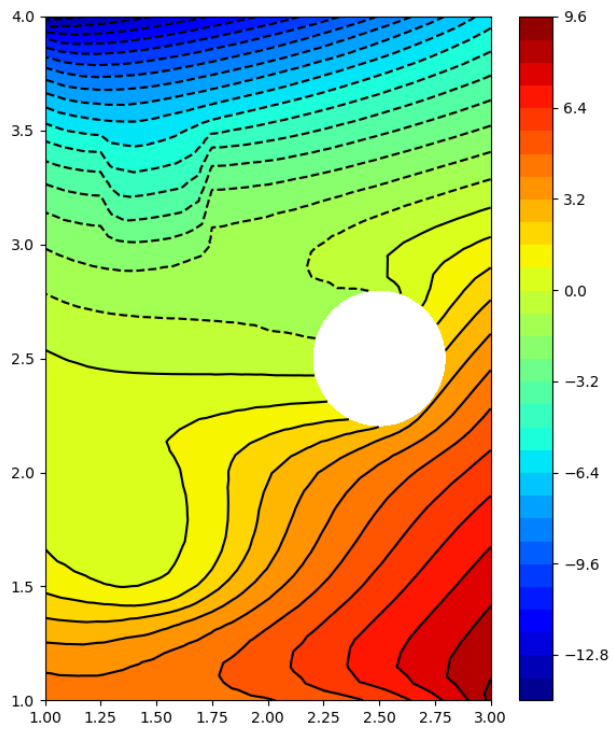
In Ileas finden Sie die Lösungen zu dem Problem:

Netz2D_p.dat enthält die (x,y)-Werte des Netzes. Laden mit `p=np.loadtxt('Netz2D_p.dat', dtype=float)`
Netz2D_t.dat enthält die Elemente des Netzes. Laden mit `t=np.loadtxt('Netz2D_t.dat', dtype=int)`
Netz2D_dr_a.dat enthält die Randknoten aus a). Laden mit `dr=np.loadtxt('Netz2D_dr_a.dat', dtype=int)`
Netz2D_dr_b.dat enthält die Randknoten aus b). Laden mit `dr=np.loadtxt('Netz2D_dr_b.dat', dtype=int)`
Netz2D_rr_b.dat enthält die Randsegmente aus b). Laden mit `rr=np.loadtxt('Netz2D_rr_b.dat', dtype=int)`
Netz2D_LoesungA.dat enthält die Lösung zu a). Laden mit `sol=np.loadtxt('Netz2D_LoesungA.dat', dtype=float)`
Netz2D_LoesungB.dat enthält die Lösung zu b). Laden mit `sol=np.loadtxt('Netz2D_LoesungB.dat', dtype=float)`
Netz2D_LoesungC.py enthält die Lösung zu c).

Das Netz können Sie sich mit `plt.triplot(p[:, 0], p[:, 1], t); plt.show()` anzeigen lassen. Vergleichen Sie nun Ihre Lösung mit der Lösung sol, z.B. mit `plt.plot(sol-IhreLoesung); plt.show()`. Der Fehler sollte kleiner als 10^{-12} sein.



Lösung a)-Teil



Lösung b)-Teil