אוניברסיטת בן גוריון
הפקולטה למדעי ההנדסה
המחלקה להנדסת מכונות

מספר קבוצה 11
מספר פרויקט 2

דו"ח קבוצתי – שיטות נומריות הנדסיות

מגישים:
אושר אזולאי 203099247
אביב אבוחצירה 203195011
ליהי קלקודה 305277071

אדר תשע"ט מרץ 2019

סעיף א' - פתרון בשיטת ניוטון ראפסון למשוואות החום הלא ליניארית, יציבה, ללא החור כאשר $k(T)=T$ (שינוי באישור).

1. פיתוח תיאורטי

בסעיף זה נפתור את משוואות החום כאשר $k(T)=T$ והמערכת נמצאת במצב המתמיד. ראשית, נתחיל מהסבר התאוריה ולאחר מכן נסביר על היישום בתוכנת MATLAB לשיטה אותה יישמנו בפרויקט זה. לבסוף, ננתח את התוצאות ונדון בהם. המשוואה בתרגיל אותה קיבלנו הינה:

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(k(T)r^2\frac{\partial T}{\partial r}\right)+\frac{1}{r^2\sin(\theta)}\frac{\partial}{\partial\theta}\left(k(T)\sin(\theta)\frac{\partial T}{\partial\theta}\right)=\frac{\partial T}{\partial t} \quad (1)$$

1.1. שיטת ניוטון ראפסון

שיטת ניוטון ראפסון הינה שיטה איטרטיבית לפתירת משוואה אלגברית לא ליניארית. כאשר פיתוחה מתבסס על קירוב ערך הפונקציה של סדרת טיילור ולכן ישנה חשיבות גבוהה לניחוש ההתחלתי.

$$y-f(x_1)=f'(x_1)(x-x_1) \quad (2)$$

כאשר $\delta x=(x-x_1)$ עבור y=0 נקבל:

$$f'(x)\big|_{x=x_1}\cdot\delta x=-f(x_1) \quad (3)$$

האיטרציה תושלם ב-3 שלבים:

1. הצבת תנאי התחלה $x=x_1$ כניחוש ראשוני
2. חישוב הנגזרת $f'(x)\big|_{x=x_1}$ וחישוב δx
3. חישוב $x_{new}=\delta x+x_{old}$

בהתאמה עבור מערכת משוואות מתקבלים הקשרים:

$$A\cdot\left[\delta T_{ij}\right]=-b \quad (4)$$

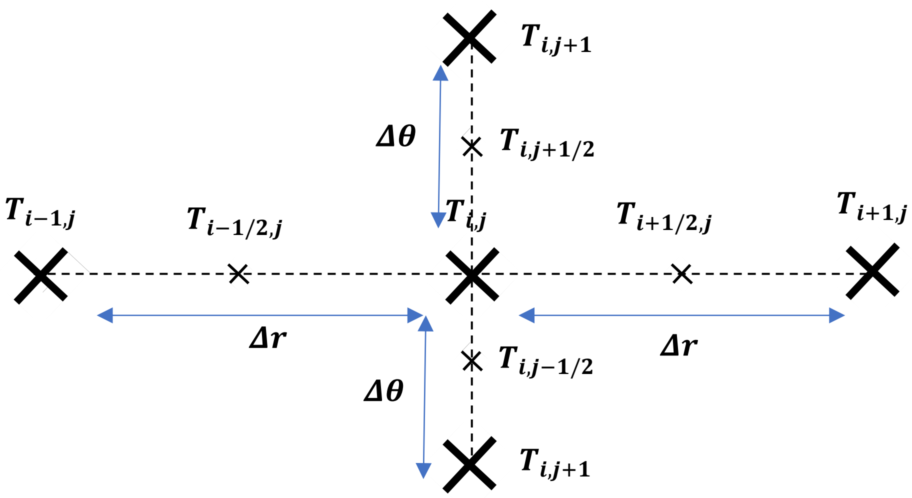
$$\left[T_{ij}\right]^{n+1}=\left[\delta T_{ij}\right]+\left[T_{ij}\right]^{n}. \quad (5)$$

כאשר A היא מטריצת היעקביאן והיא מחושבת באופן הבא :

$$A=\begin{bmatrix} \frac{\partial F_1}{\partial T_1} & \cdots & \frac{\partial F_1}{\partial T_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial T_1} & \cdots & \frac{\partial F_n}{\partial T_m} \end{bmatrix} \tag{6}$$

1.2. פיתוח ניוטון ראפסון עבור משוואת החום

לצורך מציאת מטריצת היעקביאן עבור הנקודות הפנימיות, כאשר $k(T)=T$, ניעזר בשיטת Stagger Grid . לפי שיטה זו, נשתמש בפיתוח סכמת הפרשים בדיוק מסדר שני עבור נקודות ביניים בין אינדקסי הרשת (נקודות החצי בין האינדקסים i,j). לאחר מכן, יבוצע פיתוח סכמת הפרשים סופיים נוספת בין נקודות הביניים על מנת לקבל את הפיתוח עבור הנקודה הרצויה כפי שמופיע באיור 1. באופן זה, ניתן לפתח את משוואת החום בצורה ישירה ולרוב מהירה יותר מנגזרות ישירות.



איור 1 : תיאור חלוקת הרשת לצורך פיתוח בשיטת Stagger Grid.

עבור משוואה (1) שהוצגה, נפרק את הפיתוח לשני חלקים באופן הבא :

<u>פיתוח משוואה (1) חלק א' :</u>

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(Tr^2\frac{\partial T}{\partial r}\right)=\frac{2}{r}T\frac{\partial T^n}{\partial r}+\frac{\partial}{\partial r}\left(T\frac{\partial T^n}{\partial r}\right)$$

$$\frac{2}{r}T\frac{\partial T}{\partial r}=\frac{2}{r}T_{i,j}\left(\frac{T^n{}_{i+1,j}-T^n{}_{i-1,j}}{2\Delta r}\right)$$

$$\frac{\partial}{\partial r}\left(T\frac{\partial T}{\partial r}\right)=\frac{1}{\Delta r}\left(\frac{T_{i+1,j}+T_{i,j}}{2}\cdot\frac{T_{i+1,j}-T_{i,j}}{\Delta r}-\frac{T_{i,j}+T_{i-1,j}}{2}\cdot\frac{T_{i,j}-T_{i-1,j}}{\Delta r}\right)$$

<u>פיתוח משוואה (1) חלק ב':</u>

$$\frac{1}{r^2\sin(\theta)}\frac{\partial}{\partial\theta}\left(T\sin(\theta)\frac{\partial T}{\partial\theta}\right)=\frac{1}{r^2\tan(\theta)}\left(T\frac{\partial T}{\partial\theta}\right)+\frac{1}{r^2}\frac{\partial}{\partial\theta}\left(T\frac{\partial T}{\partial\theta}\right)$$

$$\frac{1}{r^2\tan(\theta)}\left(T\frac{\partial T}{\partial\theta}\right)=\frac{T}{r^2\tan(\theta)}\left(\frac{T_{i,j+1}-T_{i,j-1}}{2\Delta\theta}\right)$$

$$\frac{1}{r^2}\frac{\partial}{\partial\theta}\left(T\frac{\partial T}{\partial\theta}\right)=\frac{1}{r^2\cdot\Delta\theta}\left(\frac{T_{i,j+1}+T_{i,j}}{2}\cdot\frac{T_{i,j+1}-T_{i,j}}{\Delta\theta}-\frac{T_{i,j}+T_{i,j-1}}{2}\cdot\frac{T_{i,j}-T_{i,j-1}}{\Delta\theta}\right)$$

עבור מצב של Steady state נקבל:

$$\frac{\partial T}{\partial t}=0 \tag{7}$$

לכן, נוכל לקבל עבור משוואה (1) מהפיתוחים שביצענו וממשוואה (7):

$$\left(\overbrace{\frac{T_{i.j}}{\Delta r\cdot r_{i.j}}+\frac{T_{i+1,j}+T_{i,j}}{2\Delta r^2}}^{b}\right)T^{n}{}_{i+1,j}+\left(\overbrace{-\frac{T_{i.j}}{\Delta r\cdot r_{i,j}}+\frac{T_{i,j}+T_{i-1,j}}{2\Delta r^2}}^{c}\right)T^{n}{}_{i-1,j}+ \tag{8}$$
$$\left(\overbrace{\frac{T_{i.j}}{2\Delta\theta\cdot r_{i.j}{}^{2}\tan(\theta)}+\frac{T_{i,j+1}+T_{i,j}}{2\Delta\theta^2\cdot r_{i.j}{}^{2}}}^{d}\right)T_{i,j+1}+\left(\overbrace{-\frac{T_{i,j}}{2\Delta\theta\cdot r^2\tan(\theta)}+\frac{T_{i,j}+T_{i,j-1}}{2\Delta\theta^2\cdot r_{i.j}{}^{2}}}^{e}\right)T_{i,j-1}+$$
$$\left(\overbrace{-\frac{T_{i+1,j}+T_{i,j}}{2\Delta r^2}-\frac{T_{i,j}+T_{i-1,j}}{2\Delta r^2}-\frac{T_{i,j+1}+T_{i,j}}{2\Delta\theta^2\cdot r^2}-\frac{T_{i,j}+T_{i,j-1}}{2\Delta\theta^2\cdot r^2}}^{a}\right)T_{i,j}=0$$

ומכאן נקבל את פונקציית $F\left(T_{i,j},T_{i+1,j},T_{i-1,j},T_{i,j+1},T_{i,j-1}\right)$ עבור נקודות הפנים של הרשת:

$$F=a_{i,j}T_{i,j}+b_{i,j}T_{i+1,j}+c_{i,j}T_{i-1,j}+d_{i,j}T_{i,j+1}+e_{i,j}T_{i,j-1}=0 \tag{9}$$

עבור תנאי שפה נוימן $F\left(T_{i,j},T_{i+1,j},T_{i-1,j},T_{i,j+1},T_{i,j-1}\right)$ המתקבלת בנקודות אלו מחושבת בעזרת נגזרות קדמיות ואחוריות :

$$F=\frac{\partial T_{ij}}{\partial \theta}=-T_{i,j+2}+4T_{i,j+1}-3T_{ij}=0 \tag{10}$$

$$F=\frac{\partial T_{ij}}{\partial \theta}=-T_{i,j-2}+4T_{i,j-1}-3T_{ij}=0 \tag{11}$$

ועבור תנאי שפה דריכלה נתאר את פונקציית $F\left(T_{i,j},T_{i+1,j},T_{i-1,j},T_{i,j+1},T_{i,j-1}\right)$ על ידי קיום הטמפרטורה בנקודות המתאימות.

$$\begin{aligned} F&=T_{i,j}(r=1,\theta)=1\\ F&=T_{i,j}(r=0.5,\theta)=0 \end{aligned} \tag{12}$$

מכאן נקבל כי בבניית מטריצת היעקוביאן קיימים שלוש סוגי שורות :

1. נקודות המתאימות לטמפרטורה עבור תנאי שפה דריכלה
2. נקודות המתאימות לטמפרטורה עבור תנאי שפה נוימן
3. נקודות פנימיות של הרשת

באיחוד כלל התנאים המופיעים במשוואות (8) - (12) נקבל הצגה של מטריצת היעקוביאן ובהתאם של וקטור הפתרונות :

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & \cdots & & & & & \cdots & 0\\
0 & -3 & \cdots & & & \cdots & 4 & \cdots & & -1 & \cdots & 0\\
\vdots & 0 & -3 & & & & \cdots & 4 & \cdots & & -1 & \vdots\\
 & & \cdots & -3 & & & & & & & & \\
 & & & & 1 & & & & & & & \\
 & & & & & 1 & & & & & & \\
 & \frac{\partial F}{\partial T_{ij+1}} & 0 & 0 & 0 & \frac{\partial F}{\partial T_{i-1,j}} & \frac{\partial F}{\partial T_{i,j}} & \frac{\partial F}{\partial T_{i+1,j}} & \cdots & \frac{\partial F}{\partial T_{ij+1}} & & \\
 & & & & & & & 1 & & & & \\
 & & & & & & & & & \vdots & & \vdots\\
\vdots & & & & & & & & & \cdots & -3 & 0\\
0 & \cdots & & & & & & & & \cdots & 0 & 1
\end{bmatrix}\cdot\left[\delta\bar{T}_{i,j}\right]=\begin{bmatrix}
T_{i,j}\\
-T_{i,j+2}+4T_{i,j+1}-3T_{i,j}\\
\vdots\\
\\
T_{i,j}-1\\
T_{i,j}\\
F(i,j)\\
\vdots\\
\\
\\
\\
T_{i,j}-1
\end{bmatrix}$$

ניתן לראות שקיימת חוקיות במטריצה ולכן ניתן לבנות אותה ואת הפתרון וכך למצוא את וקטור $\left[\delta T_{ij}\right]$.

עבור המקומות במטריצה המכילים את הנגזרות של F, בוצעה גזירה באמצעות שימוש בפרמטרי syms והפעלת גזירה על ידי פונקציית jacobian הקיימת בספריות תוכנת המטלב. לאחר מכן, נבנתה פונקציה המבוססת על תוצאות הגזירה שהתקבלו. פונקציה זו מציבה את הטמפרטורות בכל איטרציה ונקראת בעבודה שלנו D_F.

1.3. בדיקת condition number למטריצה

על מנת לבדוק אם מטריצה הינה "חולה" או "בריאה", כלומר אם קשה או קל לעבוד איתה, נשתמש בחוקים של נורמות. ההגדרה של Conditional Number עבור מטריצה ריבועית:

$$cond(A) = \|A\| \cdot \|A^{-1}\|$$

אם A הינה מטריצה סינגולרית אז $cond(A) = \infty$. בנוסף, לכל מטריצה $cond(A) \geq 1$.

חישוב Conditional Number מצריך חישוב של A^{-1} אשר לעיתים דורש כוח עיבוד יקר. לעיתים יותר מהר ללכסן מטריצה מאשר לחשב את Conditional Number.

קיימת חשיבות רבה ל– Conditional Number בפתרון מערכת משוואות לינאריות. נוכיח זאת:

עבור המערכת הלינארית $A \cdot x = b$ מתקיים $\|b\| \leq \|A\| \cdot \|x\|$. נניח כי $x \neq 0, b \neq 0$ ונחלק ב $\|b\|\|x\|$:

$$\frac{1}{\|x\|} \leq \frac{\|A\|}{\|b\|}$$

נגדיר את השגיאה כ- $r = A(x - \tilde{x})$ כאשר $x, \tilde{x}$ הם הפתרון המדויק והמשוערך. בעזרת מניפולציות של נורמות ניתן להגדיר:

$$\frac{\|x - \tilde{x}\|}{\|x\|} \leq \frac{1}{\|x\|} \|A^{-1}\| \|r\| \leq \frac{\|A\|}{\|b\|} \|A^{-1}\| \|r\| = cond\left(A\right) \frac{\|r\|}{\|b\|}$$

שינוי קטן בערכי המטריצה A או בערכי וקטור אגף ימין b, נקבל שינוי גדול גם בערך $cond(A)$ ולכן גם שינוי גדול בערך x. לכן בעייתי לפתור מטריצות שהינן "חולות" כי תהיה בעיה בהתכנסות.

קיימים מספר קריטריונים לבחינת קצב התכנסות עבור שיטות איטרטיביות. נשתמש בקריטריון מסדר שני עבור משוואה ליניארית בהתאם למשוואה (13):

$$\lim_{k \to \infty} = \frac{|P_{k+1} - P|}{|P_k - P|^q} < \lambda, P_k \neq P \tag{13}$$

כאשר λ הינה השגיאה האסימפטומטית , P הפתרון המדויק ו- q סדר ההתכנסות.

2. פיתוח סימולציה

בסעיף זה נציג את הגישה והשלבים איתם כתבנו את הקוד בתוכנת המטלב.

2.1. הפונקציות שנכתבו לצורך פתרון הבעיה

1. jacobi_syms – מקבלת $dr, d\theta$ (דלתא של הזווית ודלתא של הרדיוס) ומחשבת את היעקוביאן עבור הפונקציה F על ידיי גזירה ושימוש בפונקציית jacobian (מבוצעת פעם אחת).
2. D_F - מציבה את ערכי הטמפרטורות בכל איטרציה בנגזרות של פונקציה F לפי היעקוביאן שחושב.
3. CreateMat – מקבלת n (גודל הרשת) ואת כלל הנגזרות שחושבו ובונה את מטריצה A בצורת sparse.
4. CreateRHS – מקבלת את n, T_n, F ומחזירה את RHS שהינו וקטור אגף צד ימין.
5. Normalizing – פונקציה המנרמלת את מטריצת היעקוביאן ווקטור אגף צד ימין לצורך פישוט החישוב והקטנת conditional number.

טבלה 1 : תיאור המשתנים בסימולציה

סימון	משתנה
i, j	אינדקס מיקום במטריצה
dt	דלתא זמן
norm	נורמה של שגיאת הטמפ' מאיטרציה קודמת
r	מטריצת הרדיוס
θ	מטריצת הזווית
dr	הפרש רדיוס
$d\theta$	הפרש זווית
A	מטריצת היעקוביאן
Tn	מטריצת הטמפ'
RHS	מטריצת איברים צד ימין
T_old	מטריצת הטמפ' באיטרציה קודמת
dTn	מטריצת שינוי הטמפ'
e	מטריצת השגיאה
DF_T_ij, DF_T_iplus1j…	מטריצות נגזרות חלקיות בכל נקודה

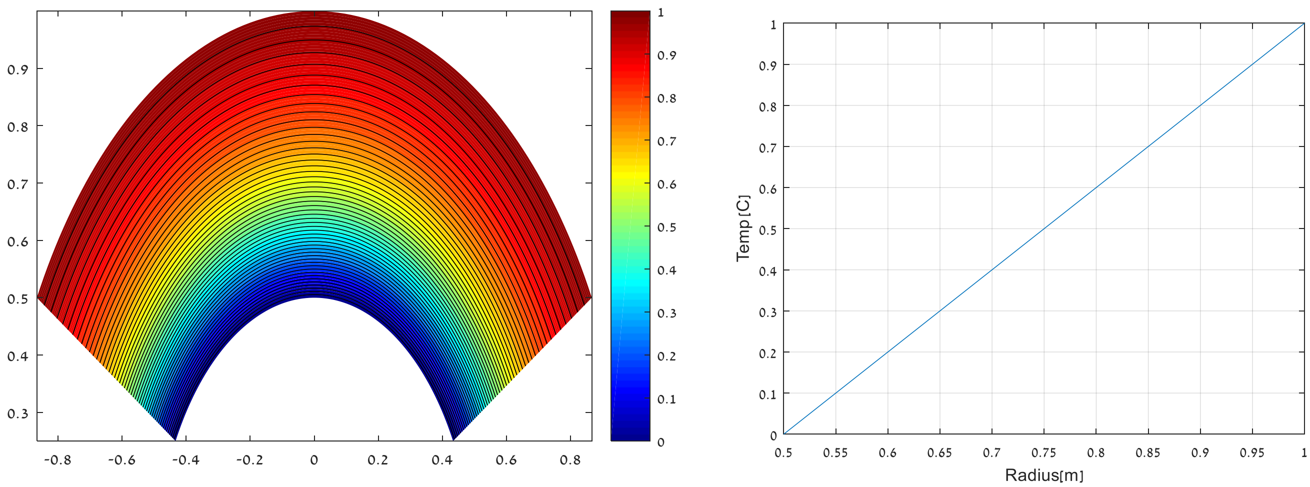
2.2. פסאודו קוד

1. אתחול תנאי הבעיה
2. כל עוד הנורמה גדולה מ 10^{-4} , בצע
   2.1. הצבת נתוני הרשת עבור $F\left(T_{i,j}, T_{i+1,j}, T_{i-1,j}, T_{i,j+1}, T_{i,j-1}\right)$ ונגזרותיו.
   2.2. יצירת מטריצה היעקוביאן - A .
   2.3. יצירת וקטור צד ימין RHS.
   2.4. נרמול מטריצה A ווקטור RHS.
   2.5. פתרון מערכות המשוואות $A \cdot \left[\delta T_{ij}\right] = -b$
   2.6. עדכון מטריצת הטמפ' עם שינוי הטמפ' בכל נקודה
   2.7. חישוב נורמה

- כלל הכתיבה בוצעה ללא לולאות אלא באופן וקטורי כדי ליעל את הקוד. בבניית הסימולציה השתמשנו בפונקציית mldivide שהינה יעילה מאוד לפתרון מטריצות sparse ומפעילה אלגוריתם פתרון שונה בהתאם למבנה המטריצה. תפקידה הינו לפתור את המשוואה $A \cdot x = b$ עבור המשתנה x ולהחזירו. כלומר, נוכל לקבל את $\left[\delta T_{ij}\right]$.
- הוספת הנרמול למטריצה A ובעקבות כך גם לוקטור RHS נבעה כתוצאה מ-cond גבוה עבור מטריצה A. מטריצה בעלת cond-number גבוה הינה "חולה", כלומר קשיחה, ולכן לעיתים נוצרת בעייתיות בהתכנסות כפי שהוסבר בפרק התיאורטי. על ידי יצירת נרמול יעיל ללא לולאות ניתן לקבל מטריצה בעלת cond בהפחתה של כמה סדרי גודל ובנוסף לא להשפיע על יעילות האלגוריתם לרעה.
- להצגת הפתרון השתמשנו בקורדינטות פולריות ובפונקציית contourf כדי לאפשר רמת צבע שונה לטמפרטורה שונה כפי שנדרש.

3. דיון בתוצאות

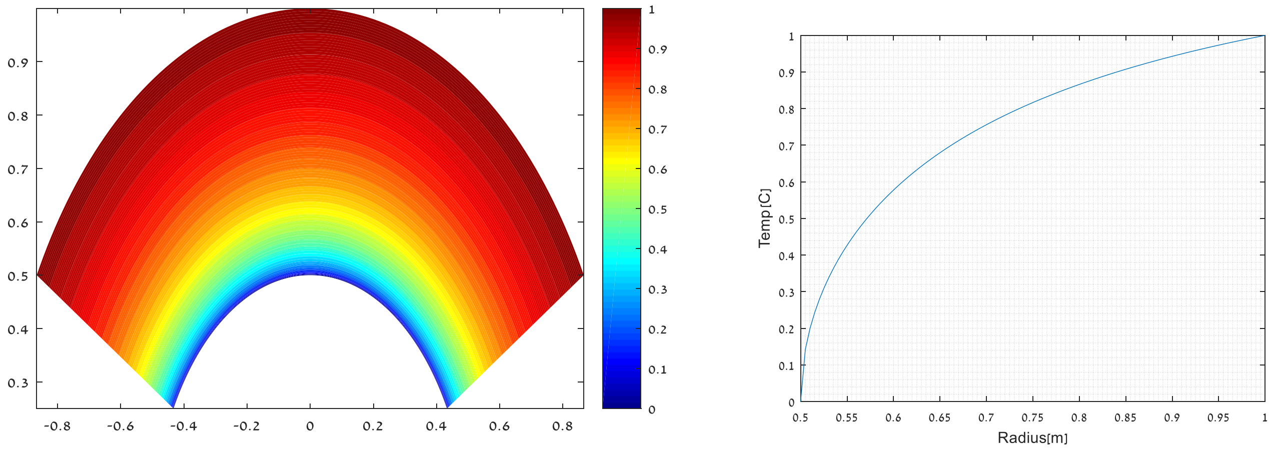
תנאי ההתחלה כפי שניתנו מוצגים באיור 2.



איור 2: פילוג טמפרטורה לינארי של תנאי התחלה, (א) עבור זווית מסויימת (כל זווית), (ב) התפלגות פולארית כפונקציה של הזווית והרדיוס.

עבור תנאי התחלה, ניתן לראות שהצבנו טמפרטורות בפילוג לינארי, זאת מכיוון שידוע שלפי ניוטון ראפסון עבור תנאי התחלה מסויימים, פילוג הטמפרטורה עלול להתבדר.

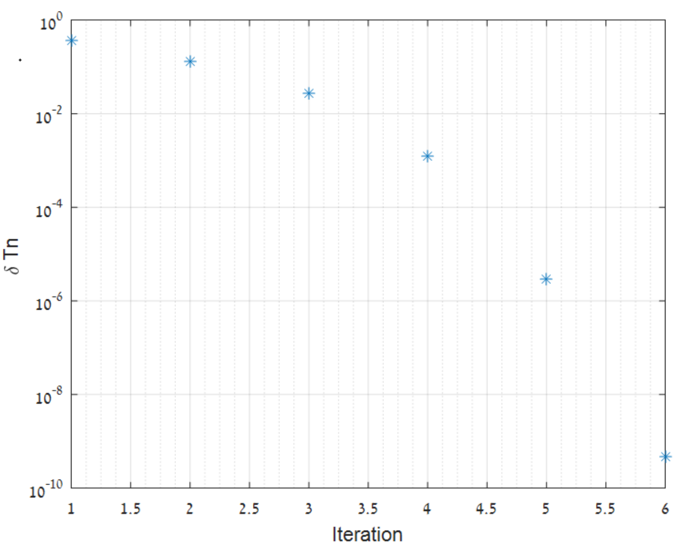
לאחר 6 איטרציות והגעה לנורמה של 10^{-5} עבור 100 נקודות נקבל כפי שניתן לראות באיור 3 :



איור 3: פילוג הטמפרטורות בטבעת לאחר סיום האיטרציות, (א) עבור זוית מסויימת, (ב) התפלגות פולארית כפונקציה של הזוית והרדיוס.

לפי איור 3, ניתן לראות שפילוג הטמפרטורה לאורך הרדיוס איננו ליניארי. ניתן לראות כי הטמפרטורה עולה בהתאם לעליית הרדיוס עם סימטריה סביב הזווית. בנוסף ניתן לראות כי מעבר החום מהשפה החיצונית גדול יותר מהשפה הפנימית, דבר זה עולה בהתאמה עם מקדם המכפלה $k(T)=T$ מכיוון שהטמפרטורה גבוהה יותר ככל שהרדיוס גדול יותר.

השתמשנו בשיטת ניוטון ראפסון שהינה שיטה שתלויה בתנאי ההתחלה. לפי עקרון פעולתה היא יכולה להתבדר עבור תנאי התחלה שאינם קרובים לפתרון המערכת משום שהיא פותחה בעזרת טור טיילור. שיטה זו שימושית מאוד במידה וקיים שורש פשוט. בנוסף, שיטה זו בעלת קצב התכנסות מסדר שני, נבחן זאת בעזרת נורמה של שינוי הטמפרטורה בכפי שמופיע איור 4, נציין כי ה- δT_{ij} שאליו התייחסנו הינו שינוי הטמפרטורה המקסימלית בנקודה כלשהיא המתקבל ברשת עבור האיטרציה.



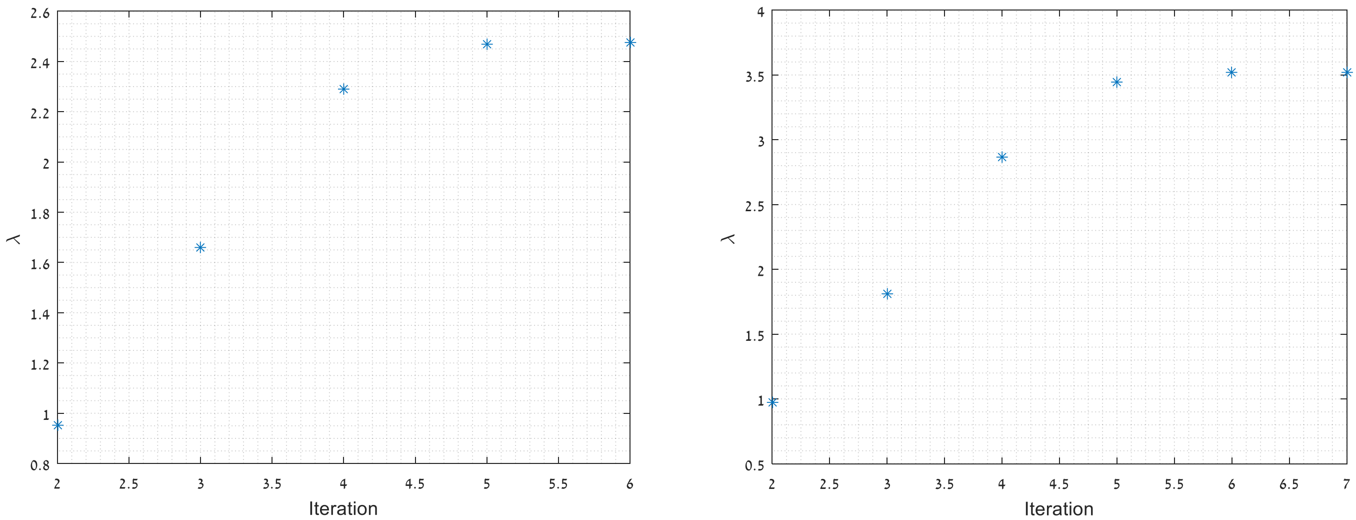
איור 4: התאמת הנורמה של הטמפרטורות לפולינום מסדר שני

ניתן לראות כי שינוי הטמפרטורה δT_{ij} קטן בצורה ניכרת מאיטרציה לאיטרציה ועל מנת לבחון את סדר ההתכנסות ניעזר במשוואה (13) שהוצגה בסעיף 1 (הסעיף התאורטי).

לכן, נבדוק התכנסות עבור משוואת החום. נבחר את הפתרון המדויק להיות הנורמה אותה הגדרנו כקריטריון העצירה וסדר התכנסות 2. מכאן מתקבל הקשר:

$$\lambda^{k\to n}{}_{k} = \max_{ij} \frac{\left|T^{k+1}{}_{ij} - T^{n}{}_{ij}\right|}{\left|T^{k}{}_{ij} - T^{n}{}_{ij}\right|^2} \cong \max_{ij} \frac{\left|\delta T^{k+1}{}_{ij}\right|}{\left|\delta T^{k}{}_{ij}\right|^2}$$

נציג את ההתייחסות בתיאור הטמפרטורה בין איטרציה קודמת לנוכחית עבור שתי רשתות שונות לבחינה.



איור 5: תיאור יחס הפרש הטמפרטורות באיטרציה k+1 לאיטרציה קודמת עבור משוואת החום עבור רשת שונות ([100,100] ימין , [50,50] שמאל).

מאיור 5 ניתן לראות כי ההתכנסות אכן מסדר שני ושמתקבל חסם לשגיאה האסימפטומטית התלוי בגודל הרשת אותה אנו פותרים, כאשר האסימפטוטה המתקבלת גבוהה יותר ככל שהרשת גדולה יותר.

4. סיכום ומסקנות

בסעיף זה, ביצענו פיתוח למשוואת חום בקורדינטות פולריות באמצעות שיטת ניוטון ראפסון. המשוואה שקיבלנו יציבה במצב המתמיד, כללה תנאי שפה מסוג דיריכלה ונוימן. פיתחנו אלגוריתם להצגת סכמת הפתרון המתקבל ביישום שיטת ניוטון ראפסון בתוכנת המטלב וניתחנו את התנהגותה עבור צעדי זמן שונים. בבחינת התוצאות שהתקבלו בפרויקט זה, קיבלנו כי שיטת ניוטון ראפסון הינה כלי יעיל לפתרון משוואה שאיננה לינארית, כאשר קצב ההתכנסות המתקבל הינו מסדר שני, אך עם זאת ישנה חשיבות רבה לתנאי ההתחלה בו בחרנו בעקבות כך שבפיתוח השיטה מתבססים על טור טיילור סביב תנאי ההתחלה. מניתוח המצב המתמיד נמצא כי הטמפרטורה ברשת מתפלגת בצורה שאיננה לינארית, ושקצב מעבר החום מהשפה החיצונית נמצא גבוהה יותר מהשפה הפנימית וזו נמצא בהתאמה למקדם המכפלה $k(T)=T$.

סעיף ב' - שיטת יעקובי לפתרון משוואה ליניארית, לא יציבה, עם החור כאשר $k(T)=1$. הצגת הפתרון עבור t=1,2,10. בחינת צעדי זמן שונים ודיון בהשפעה על היציבות והדיוק של השיטה.

1. פיתוח תיאורטי

בסעיף זה נפתור את משוואות החום כאשר $k(T)=1$, המערכת איננה במצב מתמיד וקיים חור. ראשית, נתחיל מהסבר התאוריה ולאחר מכן נסביר על היישום בתוכנת MATLAB לשיטה אותה יישמנו בפרויקט זה. לבסוף, ננתח את התוצאות ונדון בהם. המשוואה בתרגיל אותה קיבלנו הינה:

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(k(T)r^2\frac{\partial T}{\partial r}\right)+\frac{1}{r^2\sin(\theta)}\frac{\partial}{\partial\theta}\left(k(T)\sin(\theta)\frac{\partial T}{\partial\theta}\right)=\frac{\partial T}{\partial t} \tag{1}$$

1.1. שיטת יעקובי

זוהי שיטה איטרטיבית לפתרון מערכת משוואות המתכנסת מתוך ניחוש התחלתי כלשהו, לא תמיד השיטה תתכנס. האיטרציות מבוצעות עד להתכנסות. בהתאם לשיטה זו:

$$x_i^{k+1}=\frac{1}{a_{ii}}\left(b_i-\sum_{j\neq i}a_{ij}x_j^k\right), i=1,2...n \tag{2}$$

על מנת שתובטח התכנסות (תנאי מספיק) עבור כל ניחוש התחלתי חייב להתקיים:

$$\left|a_{ii}\right|>\sum_{j\neq i}\left|a_{ij}\right| \tag{3}$$

במידה והתנאי אינו מתקיים, לא ניתן לקבוע את התכנסות הסכמה.

1.2. פיתוח יעקובי עבור משוואת החום

נשתמש בשיטת ההצבה הישירה ונציב $k(T)=1$ ב- (1) עבור המשוואה הלא יציבה ונקבל:

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial T}{\partial r}\right)+\frac{1}{r^2\sin(\theta)}\frac{\partial}{\partial\theta}\left(\sin(\theta)\frac{\partial T}{\partial\theta}\right)=\frac{\partial T}{\partial t} \tag{4}$$

$$\frac{2}{r}\frac{\partial T}{\partial r}+\frac{\partial^2 T}{\partial r^2}+\frac{1}{r^2\tan(\theta)}\frac{\partial T}{\partial\theta}+\frac{1}{r^2}\frac{\partial^2 T}{\partial\theta^2}=\frac{\partial T}{\partial t}$$

לכן בפיתוח הנגזרות באמצעות נגזרות קדמיות ואחוריות מסדר שני נקבל עבור משוואה (4):

$$\frac{2}{r}\left(\frac{T_{i+1,j}-T_{i-1,j}}{2\Delta r}\right)+\left(\frac{T_{i+1,j}-2T_{i,j}+T_{i-1,j}}{\Delta r^2}\right)+\frac{1}{r^2\tan\theta}\left(\frac{T_{i,j+1}-T_{i,j-1}}{2\Delta\theta}\right)+\frac{1}{r^2}\left(\frac{T_{i,j+1}-2T_{i,j}+T_{i,j-1}}{\Delta\theta^2}\right)=\frac{3T_{i,j}^{n+1}-4T_{i,j}^{n}+T_{i,j}^{n-1}}{2\Delta t} \tag{5}$$

לאחר סידור המשוואה נוכל להגדיר את סכמת יעקובי כאשר החישוב יבוצע על ידי איטרציות בזמן, n, ואיטרציות של משתנה k:

$$\overbrace{\left(\frac{3}{2\Delta t}+\frac{2}{\Delta r^2}+\frac{2}{r^2\Delta\theta^2}\right)}^{a}T_{i,j}^{k+1,n}=\overbrace{\left(\frac{1}{r\Delta r}+\frac{1}{\Delta r^2}\right)}^{b}T_{i+1,j}^{k,n}+\overbrace{\left(-\frac{1}{r\Delta r}+\frac{1}{\Delta r^2}\right)}^{c}T_{i-1,j}^{k,n}+$$
$$\underbrace{\left(\frac{1}{2\Delta\theta r^2\tan\theta}+\frac{1}{r^2\Delta\theta^2}\right)}_{d}T_{i,j+1}^{k,n}+\underbrace{\left(-\frac{1}{2\Delta\theta r^2\tan\theta}+\frac{1}{r^2\Delta\theta^2}\right)}_{e}T_{i,j-1}^{k,n}-\underbrace{\left(\frac{-4T_{i,j}^{k,n-1}+T_{i,j}^{k,n-2}}{2\Delta t}\right)}_{RHS} \tag{6}$$

$$i,j=1\ldots n$$

$$a_{i,j}T_{i,j}^{k+1,n}=b_{i,j}T_{i+1,j}^{k,n}+c_{i,j}T_{i-1,j}^{k,n}+d_{i,j}T_{i,j+1}^{k,n}+e_{i,j}T_{i,j-1}^{k,n}-RHS_{i,j}$$

לכן, מתוך משוואה (6) נוכל לחלץ את הטמפרטורה באיטרציה הבאה של k+1:

$$T_{i,j}^{k+1,n}=\frac{b_{i,j}T_{i+1,j}^{k,n}+c_{i,j}T_{i-1,j}^{k,n}+d_{i,j}T_{i,j+1}^{k,n}+e_{i,j}T_{i,j-1}^{k,n}-RHS_{i,j}}{a_{i,j}} \tag{7}$$

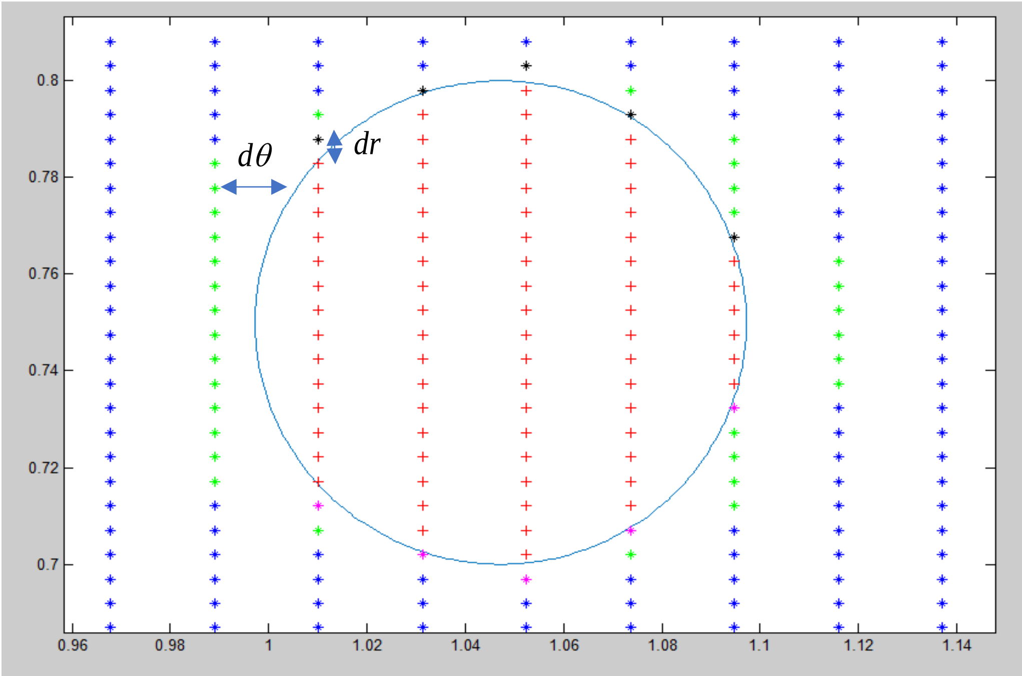
כאשר מתקיים:

$$T^{n-1,k},T^{n,k}=T_0$$

$$T^{k+1}=f(T^k)$$

2. פיתוח סימולציה

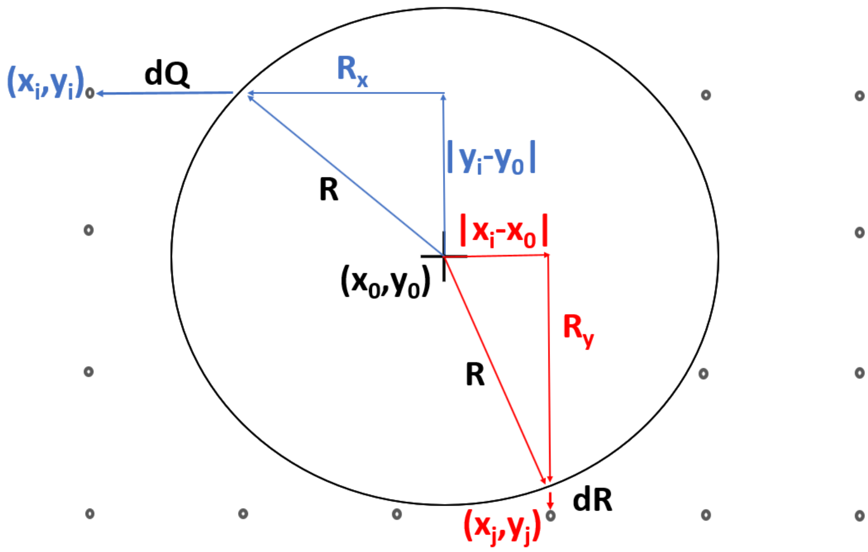
לפני ביצוע איטרציות החישוב, נבצע הגדרה לתנאי ההתחלה בתרגיל שהינם הפתרון לסעיף א'. בנוסף, נגדיר חור בהתאם למשימה. אנחנו פותרים את משוואת החור כאשר הטמפרטורה על גבולות החור מוגדרת $T=1$ ולכן בנקודות שמסביב לחור נפתור סכמה לא אחידה. עבור הגדרת החור בנינו שתי מטריצות שתפקידן הוא שמירת המרחקים בין נקודות עבור הרדיוס והזווית ($dr_{ij}, d\theta_{ij}$). האלגוריתם עובר על כל הנקודות ומגדיר את המרחקים בשתי המטריצות עבור החור. על מנת לבדוק, הצגנו וסימנו בצבע, כפי שניתן לראות באיור 5, את הנקודות שזקוקות לאופן חישוב מסוים כאשר מסווגות לנקודות בעלות dr ונקודות בעלות $d\theta$ שונה (כאשר קיימות נקודות שמתאימות לשני המקרים). ההתייחסות באיור הינה לרשת קרטזית מטעמי נוחות המחשה אך היישום הסופי הינו על רשת פולרית.



איור 5: תצוגת הנקודות המקיפות את החור ובדיקת מרחקם מהמעגל

ניתן לראות לפי איור 5 את חלוקת הנקודות לסוגים שונים. הנקודות הירוקות קיבלו חישוב במטריצה $d\theta$ כי יש להן מרחק אופקי מהמעגל ואילו הנקודות השחורות והורודות קיבלו חישוב במטריצה dr כי יש להן מרחק אנכי מהמעגל (קיימות נקודות אשר להן התייחסות גם ב- $d\theta$ וגם ב- dr אשר אינן מוצגות באיור).

נדגים את אופן החישוב עבור מרחקי dr ו- $d\theta$ באמצעות איור 6:



איור 6: אופן חישוב הפרשי המרחקים עבור הרדיוס והזווית

החישוב מבוצע באופן גאומטרי באמצעות:

$$R_x=\sqrt{R^2-(y_i-y_0)^2}\qquad R_y=\sqrt{R^2-(x_j-x_0)^2}$$
$$dQ=|x_i-x_0|-R_x\qquad dR=|y_j-y_0|-R_y$$

2.1.הפונקציות הנוספות שנכתבו לצורך פתרון הבעיה

1. creating_circle – תפקידה הינו יצירת חור לטובת פתרון רשת לא אחידה. היא מקבלת את כל הנקודות שמסביב ובתוך החור, ומחזירה מרחקים בתוך מטריצות בכיוון R ו-Q של כל נקודה ביחס לאחרות או ביחס לשפת המעגל, מה שיותר קרוב.
2. Calc_Coefficient – מקבלת ערכים של כל המשתנים ומחשבת את המקדמים הקבועים במשוואת החום.

טבלה 1 : תיאור משתנים בסימולציה

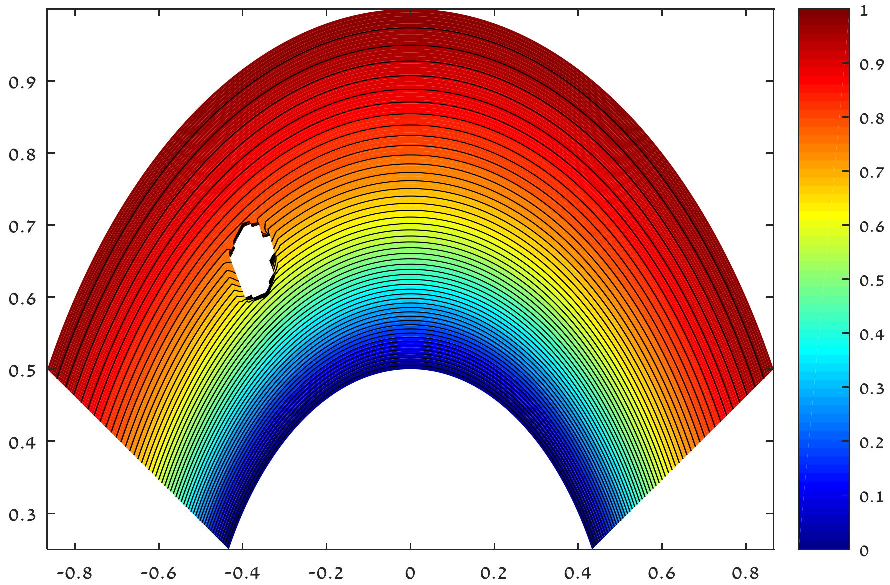
סימון	משתנה
i, j	אינדקס מיקום במטריצה
dt	דלתא זמן
norm	נורמה של שגיאת הטמפ' מאיטרציה קודמת
r	מטריצת הרדיוס
θ	מטריצת הזווית
dr	מטריצת הפרש מרחק הרדיוס מנקודות סמוכות
$d\theta$	מטריצת הפרש מרחק הזווית מנקודות סמוכות
c,d,f,p	מטריצת נקודות סמוכות חור לצורך חישוב סכמה לא אחידה
One, two...five	מקדמי המשוואה לצורך פתרון בסכמת יעקובי
Tn	מטריצת הטמפ'
RHS	מטריצת איברים צד ימין
Tn_check	מטריצת הטמפ' באיטרציה קודמת
T _old	מטריצת הטמפ' עבור צעד זמן קודם
T _old_old	מטריצת הטמפ' עבור 2 צעדי זמן קודמים

2.2.פסאודו קוד

1. אתחול תנאי הבעיה
2. יצירת חור עם דופן בטמפ' אחידה ויצירת מטריצה DR,DQ המכילות את המרחקים בן נקודות סמוכות לצורך פתרון בעיית רשת לא אחידה בנקודות רשת הסמוכות לחור
3. חישוב מקדמים קבועים במשוואת החום
4. מ-t=0 עד ל t_end:
   4.1. שמירת מטריצת הטמפ' בזמן n-1,n-2
   4.2. חישוב וקטור איברים RHS
   4.3. כל עוד הנורמה גדולה מ 10^{-4} , בצע:
      4.3.1.קיום דרישת תנאי שפה נוימן
      4.3.2.פתרון משוואת החום בעזרת שיטת יעקובי .
      4.3.3.חישוב נורמה

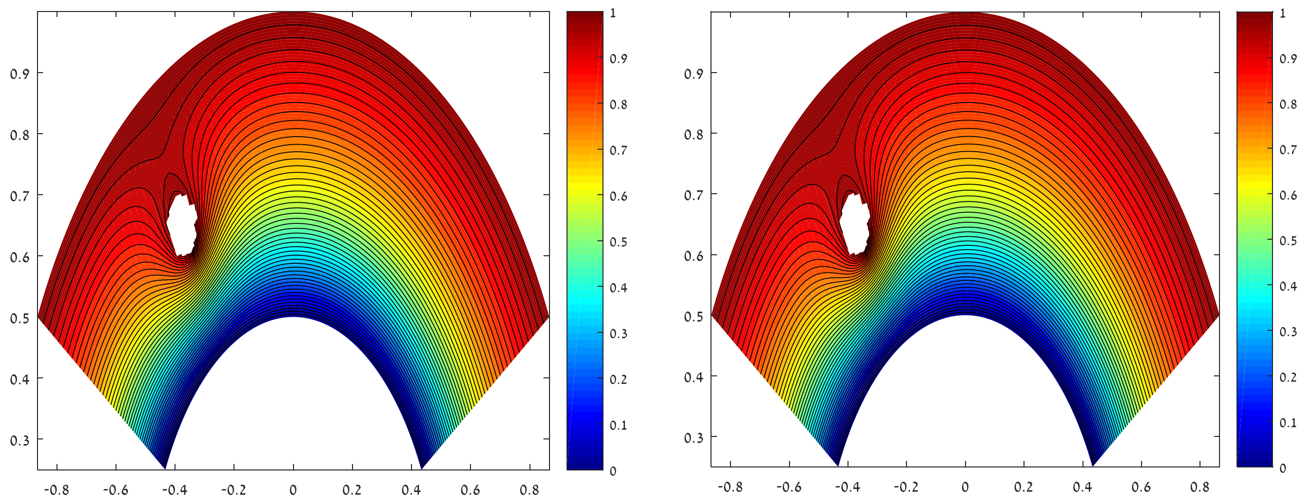
3. דיון בתוצאות

ראשית, בוצעה בחינה עבור איטרוולי זמן שונים. תנאי התחלה עם אינטרוול זמן של מאית השנייה :



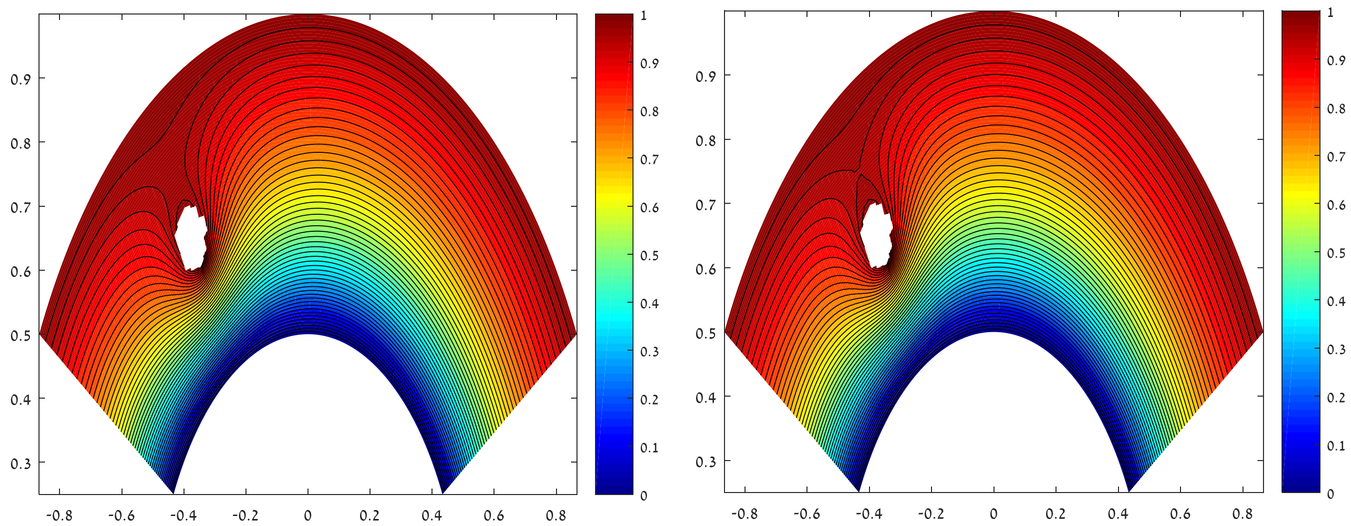
איור 7: התפלגות הטמפרטורה ברשת עבור t=0 [sec] כאשר dt=0. 01[sec]

התפלגות הטמפרטורה כעבור 0.05 שניות ושנייה אחת עם אינטרוול זמן של מאית השנייה :



איור 8: התפלגות הטמפרטורה ברשת כאשר dt=0. 01[sec], מימין עבור t=0.05[sec], משמאל עבור t=1[sec]

התפלגות הטמפרטורה כעבור 2,10 שניות כאשר אינטרוול זמן הינו מאית השנייה:



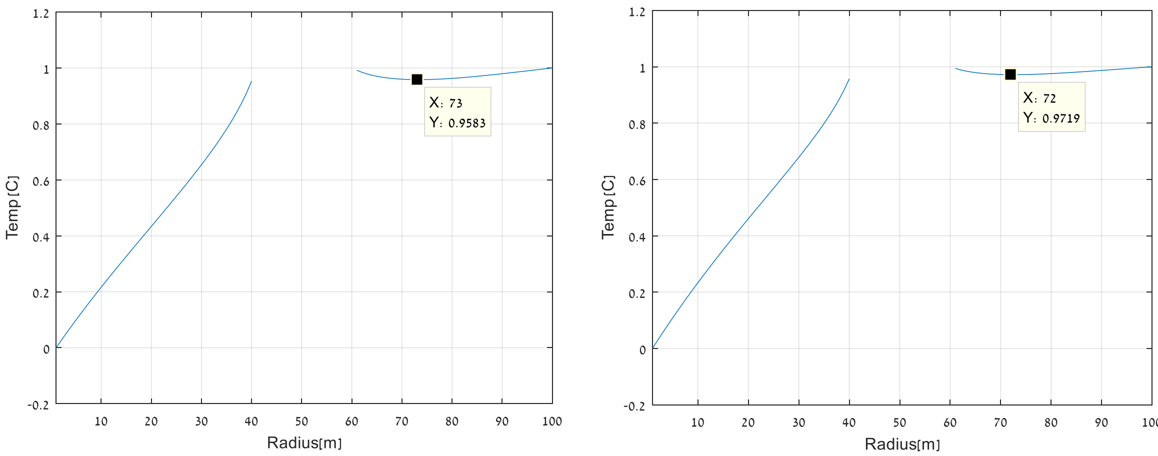
איור 9: התפלגות הטמפרטורה ברשת כאשר dt=0. 01[sec], מימין עבור t=2[sec], משמאל עבור t=10[sec]

עבור כל צעד זמן מתבצעות מספר איטרציות עד להגעה להתכנסות לנורמה שנקבעה. בבדיקה שערכנו, חילצנו את מספר האיטרציות עבור כל צעד זמן בהשוואה לצעדי זמן שונים כפי שניתן לראות בטבלה 1. נראה כי בצעדי הזמן הראשונים מתבצעות יותר איטרציות ביחס לשאר עד להתכנסות – מגמה זו נראית עבור כל אינטרוול זמן שנבחר. מכאן נסיק כי ההגעה למצב המתמיד מתקבלת בשלב דיי מוקדם בהתאמה לאיור 8 מאחר ושינוי הטמפרטורה בכל צעד זמן פוחת עם ההתקדמות בזמן האופן ניכר. בנוסף, קיבלנו שעבור צעדי זמן גדולים יותר, נקבל פתרון המתכנס באמצעות יותר איטרציות בצעדי הזמן הראשונים ומכאן נסיק כי דיוק תיאור תהליך הפתרון לאורך השתנות הזמן משתפר עם בחירת אינטרוולי זמן קטנים יותר.

טבלה 2: השוואת מספר איטרציות בצעדי זמן שונים.

צעד זמן [sec] / מספר איטרציית זמן	1	0.1	0.05	
1	2224	904	574	מספר איטרציות
2	805	858	588	
3	1	642	513	
4	1	395	418	
5	1	186	311	
6	1	70	230	
7	1	18	165	
8	1	1	116	
9	1	1	80	
10	1	1	54	

בנוסף, בהתאם לאיור 10 ניתן לראות שככל שצעד הזמן גדול יותר, החור משפיע פחות לעומת צעד זמן קטן יותר מאחר וניתן לראות שבסביבת החור הטמפרטורה משתנה באופן יותר משמעותי כאשר צעדי הזמן קטנים. את ההבדל הצגנו באופן גרפי באיור 10, בתצוגה של פילוג הטמפרטורות עבור זווית בה עובר מרכז החור.



איור 10: התפלגות הטמפרטורה ברשת עבור t=10[sec] בזווית מרכז החור, מימין dt=0.0001[sec] ומשמאל dt=0.1[sec]

אפשר לראות מאיורים 7-9 שהשיטה תמיד מתכנסת והינה יציבה ללא תנאי, אך איננו יכולים לבדוק את תנאי היציבות בעזרת הכלים שנלמדו בקורס. הסיבה לכך הינה שתנאי השפה בתרגיל שפתרנו אינם מחזורים ולכן לא נוכל להוכיח זאת בצורה חד משמעית.

בנוסף לכך, ביצענו בדיקה לתנאי ההתכנסות בשיטת יעקובי בהתאם למשוואה (3) כפי שהוצגה בסעיף 1 (חלק תאורטי) עבור מספר אינטרוולי זמן שונים :

טבלה 3 : בחינת תנאי התכנסות יעקובי באיטרציית הזמן הראשונה עבור אינטרוולי זמן שונים.

צעד זמן [שניות]	0.01	0.01	0.0001
$\max\left(\sum_{j\neq i}\lvert a_{ij}\rvert / \lvert a_{ii}\rvert\right)$	0.995	0.994	0.9536

בבחינת צעדי הזמן השונים כפי שמופיע בטבלה 2, ניתן לראות כי התנאי המספיק להתכנסות המתאים לשיטת יעקובי מתקיים. זאת ועוד, ניתן לראות שמדד תנאי ההתכנסות קטן ככל שצעד הזמן קטן.

4. סיכום ומסקנות

בסעיף זה, ביצענו פיתוח למשוואת חום בקורדינטות פולריות באמצעות שיטת יעקובי. המשוואה שקיבלנו איננה יציבה, כללה תנאי שפה מסוג דיריכלה ונוימן וחור במיקום נתון. פיתחנו אלגוריתם להצגת סכמת הפתרון המתקבל ביישום שיטת יעקובי וניתחנו את התנהגותה עבור צעדי זמן שונים. בבחינת התוצאות שהתקבלו בפרויקט זה, קיבלנו שהסכמה מתכנסת למצב המתמיד תוך זמן מהיר מאוד (כפחות משניה) עבור צעדי זמן שונים. בבחינת האיורים שהוצגו (איורים 7-9) וגדלים של צעדי זמן ורשת שונים, קיבלנו כי האלגוריתם מתכנס תמיד והינו יציב ללא תנאי.

בנוסף, בחנו את ההבדלים בין צעדי הזמן והסקנו כי עבור אותו זמן מדידה, ככל שצעד הזמן קטן יותר, החור משפיע יותר על הטמפרטורה בסביבתו מאשר עבור צעד זמן גדול יותר ותיאור השתנות הטמפרטורה ברשת מדויק יותר ככל שנבחר צעד זמן קטן יותר. לבסוף, נבחן תנאי ההתכנסות לסכמה וראינו כי הסכמה מתכנסת עבור צעדי זמן שונים.

מקורות

1. הרצאות מתוך הקורס ״שיטות נומריות מתקדמות״, 2019.
2. Anderson, D., Tannehill, J. C. & Plecher, R.H. , “Computational fluid mechanics and heat transfer” (2016).