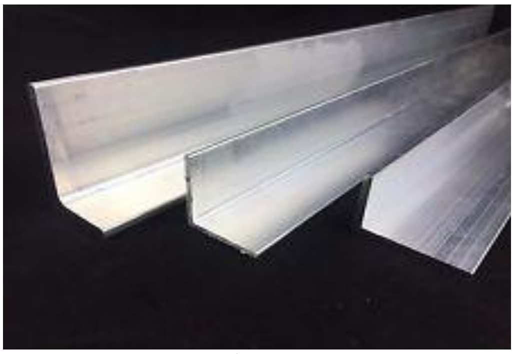
אמיר ברדה
304848336

<u>אלגוריתמים להדפסה תלת ממדית – פרוייקט סוף</u>

<u>רקע:</u>

שיטת האלמנטים הסופיים (**FEM- Finite Element Method**), היא שיטה נומרית לקבלת פיתרון מקורב ודיסקרטי למשוואות דיפרנציאליות על תחום עם תנאי שפה, כך שהמערכת שהמתקבלת הינה מוגדרת היטב. לשיטת א"ס שימוש נרחב במתמטיקה ובהנדסה. בעזרתה ניתן לקבל פתרונות לבעיות הנדסיות עבור גאומטריות ותנאי שפה מורכבים.
על מנת להפעיל את שיטת א"ס, יש צורך לבצע דיסקרטיזציה של הגאומטריה ל**אלמנטים**.
אלמנט מוגדר כפאה (face) קמורה, בעל צורה גאומטרית פשוטה (בדו מימד – לרוב משולש או מרובע, בתלת ממד – לרוב טטרהדר).
בעבודה זו נתרכז ברישות דו ממדי, שהינו פשוט יותר, אך לא פחות רלוונטי ביישומיו, מרישות תלת ממדי.
בפרט, בעולם התכן המכאני, נפוץ העובי של פריט קטן בסדר גודל ממידות האורך והרחוב שלו (ראה איור). לרוב, אנליזת א"ס מבוצעת על מנת לחשב מאמצים\תזוזות שמתקבלות עקב הפעלת עומסים.

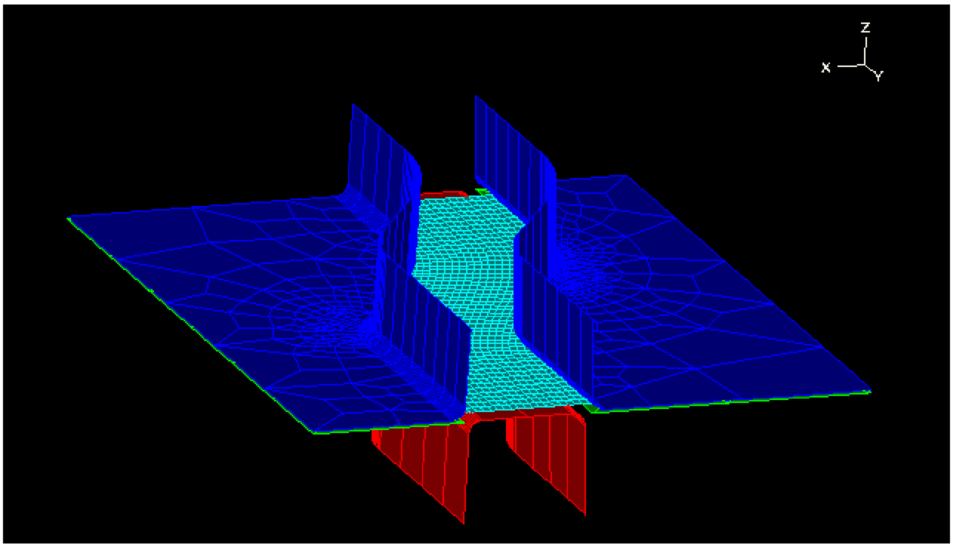


איור 1: זוויתנים משוחלים (extrusions)

במקרים כאלה, ניתן להתעלם ממימד העובי ולרשת את המשטח המרכזי (**midsurface**) באמצעות אלמנטים דו ממדיים.

ישנם שני סוגים של אלמנטים נפוצים ברישות דו ממדי: רישות משולש ורישות מרובע.

אמיר ברדה
304848336



איור 2: דוגמא לרישות מרובע (QUAD) דו ממדי של חלק עשוי מפח מכופף (sheet metal)

מתמטית, רישות מרובע עדיף על רישות משולש, עקב היכולת שלו לתפוס באופן מדוייק שינויי גרדיאנט לינאריים בערכי שדה המאמצים\תזוזות. אלמנט משולש, לעומת זאת, מסוגל מתמטית לתאר ערך קבוע של מאמץ\תזוזה בכל שטחו. לכן, ניתן לרשת משטח בפחות אלמנטים מרובעים על מנת להשיג אותה רמת דיוק אנליטית שתתקבל עבור רישות משולש צופף יותר.

בנוסף, במקרים רבים, בהנדסה מכאנית, שדות התזוזות והמאמצים רחוק מנקודות הפעלת הכוח, הם לינאריים (כתוצאה מכפיפה של הפריט, לפי תורת הכפיפה של אוילר-ברנולי).

<u>מטרת הפרוייקט:</u>

מטרת הפרוייקט היא לממש אלגוריתם לרישות Quad ממאמר עדכני ולחקור את הביצועים שלו, יחסית לפתרונות מסחריים וקוד פתוח.

אלגוריתם הרישות הוא עפ"י המאמר:
Jinwoo Choi and Yohngjo Kim , Development of a New Algorithm for Automatic Generation of a Quadrilateral Mesh, International Journal of CAD/CAM Vol. 10, No. 2, pp. 00~00 (2011).

<u>תיאור השיטה המוצעת במאמר:</u>

*קריטריון קבלה של רישות **QUAD**:*

רישות **מספק** במידה שכל הזוויות הפנימיות של האלמנטים מקיימות:

$$45^o \leq \theta_{internal} \leq 135^o \quad (1)$$

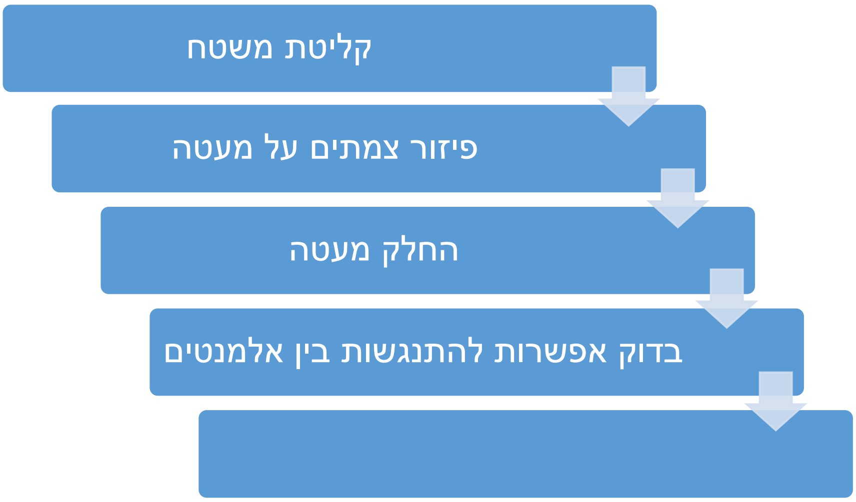
רישות יקרא **סביר** במידה וכל הזוויות הפנימיות של האלמנטים מקיימות:

אמיר ברדה
304848336

$$35^o \leq \theta_{internal} \leq 150^o \quad (2)$$

סכימה של האלגוריתם המוצע במאמר:

Commented [I1]: להוסיף תמונה ב- VISIO



אמיר ברדה
304848336

קליטת קובץ שמתאר את הגאומטריה:

במקור, המטרה הייתה לקבל קובץ **parasolid**. קובץ זה מתאר את הגאומטריה של הגוף התלת ממדית באמצעות פרמטריזציה של המשטח שמהווה את שטח הפנים שלו.
מכיוון שפרויקט זה מתעסק ברישות משטח באלמנטים ריבועים דו ממדיים, הנחת עבודה היא שקובץ ה-**parasolid** מתאר משטח דו מימדי שנמצא מישור **XY**.

המבנה של קבצי **parasolid** מתוחזק ע"י חברת **siemens** ומופץ במסמך **XT reference**. לצערי הגרסה העדכנית ביותר שנמצאה באינטרנט היא מ-2008 (**parasolid 12**), בעוד כיום קיימת גרסה **30**, ושירותי תיב"ם כמו **onShape** מאפשרים לשמור לכל הפחות בגרסת **parasolid 25**.

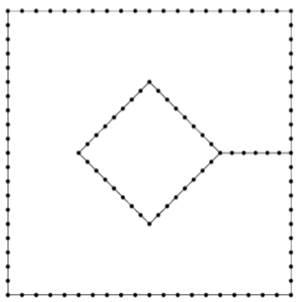
פניות ל-**siemens** דרך הערוצים המקובלים, לצורך קבלת מסמך **reference** עדכני, לא נענו במועד כתיבת מסמך זה.
מכיוון שהמטרה העיקרית של פרוייקט זה היא מימוש אלגוריתם לרישות משטחים דו ממדיים, הוחלט לממש מנגנון קלט שיקל קבלת קובץ **Parasolid** בעתיד.

התוכנית מקבלת קובץ שמכיל עקומי **NURBS** (הנחת עבודה: העקומים יוצרים משטח סגור). במידה וקיימים קדחים במשטח, יש לחבר אותם באמצעות קו ישר למשטח (פעולה זו מתבצעת באופן ידני גם במאמר, עפ"י (מאמר של **talbert**) ניתן ומתוכנן להפוך אותה לאוטומטית, כך שהמשתמש לא יצטרך להתעסק בכלל עם הגאומטריה).

הגדרת מעטה התחום:

יצירת נקודות **seed** ראשוניות על המעטפת. מתבצע באמצעות תיאור השפה כעקום רציף, כפי שתואר בפסקה הקודמת.

לדוגמא: הגאומטריה הבאה:



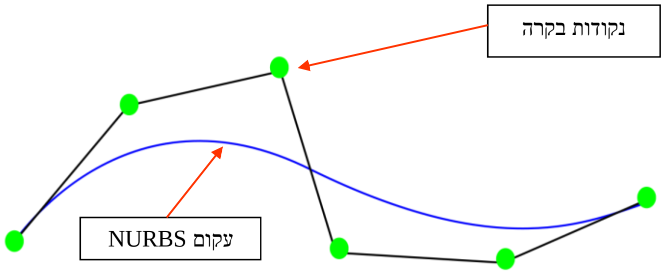
תהיה מורכבת מארבעה ישרים על השפה החיצונית:
[{(-1,-1),(1,-1)},{(1,-1),(1,1)},{(1,1),(-1,1)},{(-1,1),(-1,-1)}]
וארבעה ישרים על השפה הפנימית:
[{(-0.25,0),(0,0.25)},{(0,0.25),(-0.25,0)},{(-0.25,0),(0,0.25)},{(0,0.25),(-0.25,0)}]
בנוסף, ישנם שני ישרים שמחברים בין שני התחומים:
[{(0.25,0),(1,0)}],[{(1,0),(0.25,0}]

משטחים מוגדרים כאוסף של עקומי **NURBS** (ראה איור 3).

NURBS:

Nurbs (Non Uniform rational b-splines) הוא מודל מתמטי ליצירת והצגת עקומים ומשטחים. מודל זה בשימוש נרחב בגרפיקה ממוחשבת, ובפרט ביישומי תיב"מ, וממומש בפורמטים סטנטרדיים בשימוש ב- **CAE (Computer Aided Engineering)** כגון **IGES**, **STEP** ו-**XT**.

אמיר ברדה
304848336



איור 3: עקום NURBS

תיאור עקום **NURBS** מורכב משלושה אלמנטים:

1. נקודות בקרה (**Control Points**)
2. ווקטור קשרים (**Knot vector**)
3. משקלים (**weights**)

הגדרות **(מקור 1):**

<u>מעלה (Degree)</u>
הוא מספר שלם חיובי (לרוב 1,2,3 או 5). המעלה של עקום מתארת את צורתו. קווים ישרים **(Linear)** הם ממעלה 1. עיגולים הם ממעלה 2 (**Quadratic**), ורוב העקומים ה"כלליים" (**free-hand**) הם ממעלה 3 (**Cubic**) או 5 (**Quintic**). נקודות בקרה קובעות את הצורה של העקום. כל נקודה בעקום מחושבת ע"י סכום משוקלל של מספר נקודות בקרה (ראה ?? לפירוט).
הסדר של עקום **NURBS** הוא מספר שלם חיובי ששווה ל-(**Degree+1**).
ניתן להגדיל את המעלה של עקום **NURBS** מבלי לשנות את צורתו (יתירות בנקודות הבקרה), אך באופן כללי, לא ניתן להקטין את מעלתו של עקום **NURBS** ללא שינוי צורה.

<u>נקודות בקרה</u>
רשימה של **Degree+1** נקודות. עריכת צורתו של העקום מתבצעת ע"י שינוי מיקום\כמות נקודות הבקרה. לכל נקודת בקרה מיוחס מספר שנקרא **משקל** (לרוב מספר חיובי). כאשר לכל נקודות הבקרה יש אותו משקל, העקום נקרא לא רציונלי (**Non-rational**), אחרת, העקום נקרא רציונלי (**Rational**). בפועל, רוב עקומי ה-**NURBS** הם לא רציונליים. עקומים מסויימים, כמו עיגולים ואליפסות, הם תמיד רציונליים.

<u>קשרים (Knots)</u>
הקשרים הם רשימה של (**Degree+N-1**) מספרים, כאשר **N** הוא מספר נקודות הבקרה. רשימה זו גם נקראת **ווקטור הקשרים** (**Knot Vector).** רשימת הקשרים חוקית חייבת לקיים את התנאים הבאים:
1. מספרים ברשימה בסדר עולה (לא בהכרח מונוטוני).
2. מספר פעמים שמספר מופיע ברשימת הקשרים לא גדול יותר ממעלת העקום.

כמות הפעמים שמספר חוזר על עצמו ברשימה נקרא הדרגה של הקשר (**Knot multiplicity**). קשר בעל דרגה ששווה למעלת העקום נקרא קשר מדרגה מלאה (**Full-multiplicity**). קשר בעל דרגה 1 נראה קשר פשוט.

מודל מתמטי **(מקור 2):**
על מנת לחשב מיקום נקודת של נקודה על פני עקום **Nurbs** נדרש מעלה, נקודות בקרה ורשימת הקשרים.

$$\boldsymbol{C(u)} = \frac{\sum_{i=0}^{n} \boldsymbol{w_i \cdot P_i \cdot N_{i,k(u)}}}{\sum_{i=0}^{n} \boldsymbol{w_i \cdot N_{i,k(u)}}}$$

כאשר
$\boldsymbol{w_i}$: המשקל של נקודת בקרה **i**

אמיר ברדה
304848336

$\boldsymbol{P_i}$: נקודת בקרה **i**
$\boldsymbol{N_{i,k}}$: פונקציית בסיס של **B-spline** מדרגה **k**

פונקציות הבסיס מוגדרות רקורסיבית כ:

$$\mathrm{N}_{i,k(u)} = \frac{u - t_i}{t_{i+k} - t_i} \cdot N_{i,(k-1)[u]} + \frac{t_{i+k} - u}{t_{i+k+1} - t_{i+1}} \cdot N_{i+1,(k-1)[u]}$$

עם תנאי העצירה:

$$N_{i,0(u)} = \begin{cases} 1 & if\ t_i \le u < t_{i+1} \\ 0 & else \end{cases}$$

כש-t_i הם הקשרים ברשימת הקשרים.

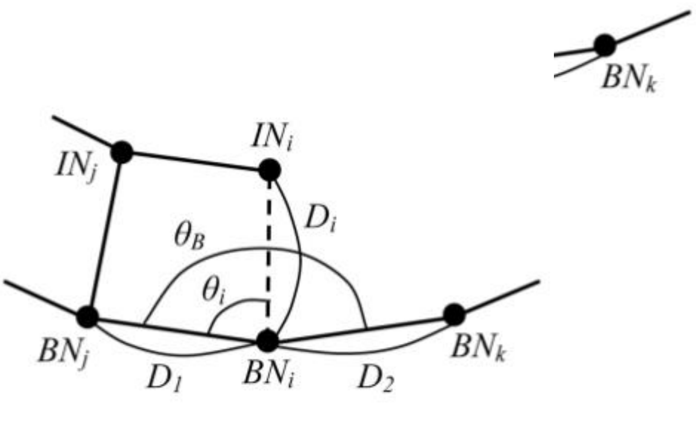
$$\mathrm{U} = \{t_0, t_1, \dots, t_m\}$$

לפונקציית הבסיס התכונות הבאות:
1. $N_{i,k(u)} \ge 0$
2. $\mathrm{N}_{i,k(u)} = 0$ אם u לא ב-$[t_i, t_{i+k+1})$
3. אם u נמצא ב-$[t_i, t_{i+1})$, אזי פונצקיות הבסיס שלא נעלמות הן $N_{i-k,k(u)}, \dots, N_{i,k(u)}$
4. $\sum_{j=i-k}^{i} N_{j,k(u)} = \sum_{i=0}^{n} N_{i,k(u)} = 1$ (תכונה זו נקראת ***partition of unity***)
5. במקרה של ערכים שחוזרים על עצמם ברשימת הקשרים, אזי יתקבלו ערכים של $\frac{0}{0}$, שיוגדרו כ-***0***.

היתרון בתיאור עקום באמצעות **NURBS**:
1. ניסוח מתמטי אחיד לצורות אנליטיות סטנדרטיות (חתכים קוניים), ולצורות "אורגניות" (**free-form**).
2. הרובוסטיות והגמישות בייצוג של צורות שונות.
3. ניתן לבצע הערכות ומניפולציות מתמטיות ביעילות.
4. השימוש בנקודות הבקרה שקובעות את צורתו של העקום, ולכן עדיין מאפשר אינטרקציה אינטואיטיבית יחסית עם משתמש אנושי לעריכת צורת העקום.
5. אינווריאנטים תחת טרנספורמציות אפיניות וכן טרנספורמציות של פרספקטיבה.
6. מהווים הכללה של **B-Splines** ועקומי **Bezier**, שהן בעצמם שיטות פופלריות לייצג מתמטית עקומים ומשטחים.

<u>הסטת צמתים ליצירת אלמנטים חדשים</u>

הסטת צמתים ליצירת אלמנטים חדשים תתבצע לכיוון שפונה הרחק משפת התחום. כמתואר באיור 4 .



אמיר ברדה
304848336

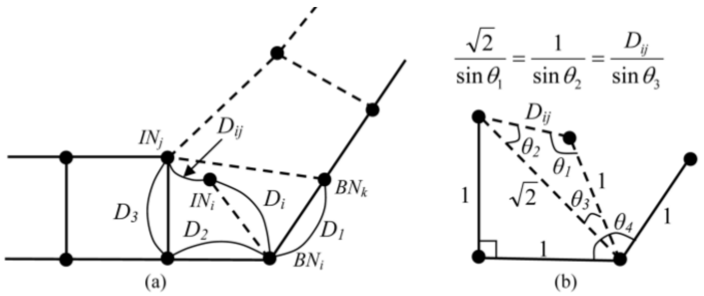
איור 4: הסטת צמתים ליצירת אלמנטים חדשים

כיוון (θ_i) ומרחק (D_1) ההסטה נקבעים ע"י הקשרים הבאים:

$$D_i = (D_1 + D_2)/2 \quad (3)$$

$$\theta_i = \theta_b/2 \quad (4)$$

מחיקת צומת שיוצרת אלמנט לא איכותי

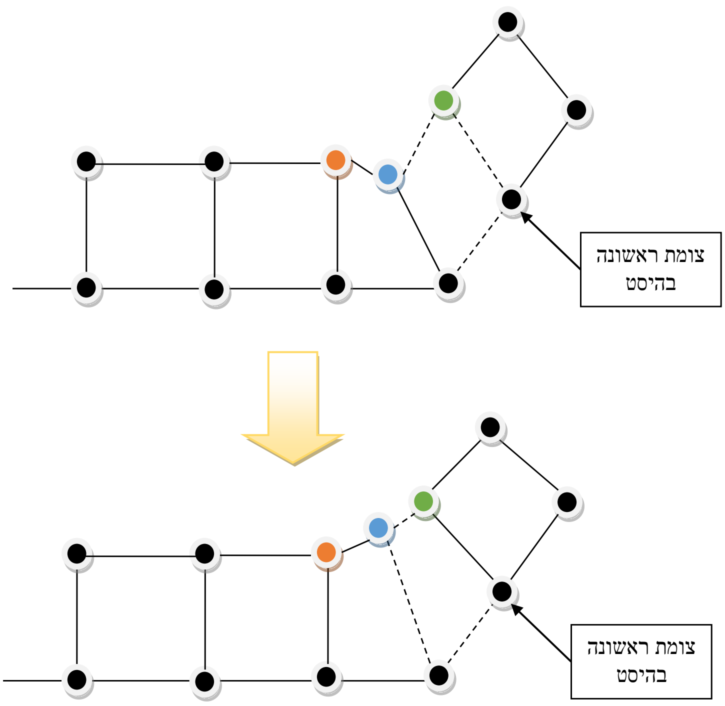


איור 5: מחיקת צמתים שיוצרים אלמנטים פגומים

בפרט, משיקולים טריגונומטריים (ראה איור 5), מתקבל שיש למחוק את הצומת אם $D_{ij} < 0.57D_i$, מכיוון שאז תתקבל זווית פנימית גדולה מ-135^o. קיימת חפיפה מסויימת בין מקרה זה ל-**case 1** *בהסרת אלמנטים לא איכותיים לאחר רישות והחלקה*. במאמר לא נתון היה כיצד להתמודד עם המקרה שבו האלמנט שיש למחוק האחרון לפני סגירת השורה, יש לאחד את שני הצמתים.

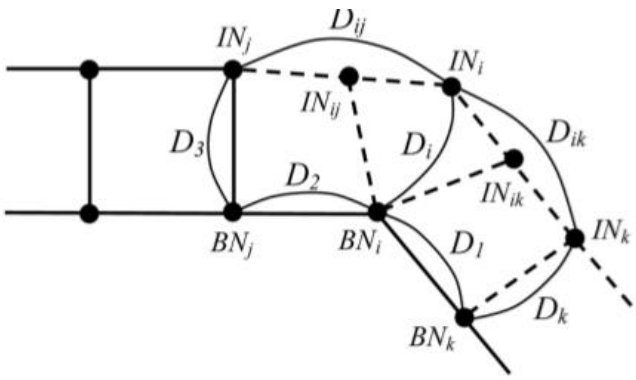
הפיתרון שנבחר להתמודדות עם המקרה הנ"ל, הוא העבר הצומת שמוסטת (מסומנת הכחול באיור 6) לנקודת הביניים בין הצומת הקודמת לצומת הראשונה שהוסטה (מסומנות בכתום וירוק, בהתאמה, באיור 6). סה"כ הזוויות הפנימיות שהתקבלו עבור שני האלמנטים שיווצרו עומדים בתנאי האיכות עפ"י ניסיונות ההרצה שבוצעו.

אמיר ברדה
304848336



איור 6: טיפול בצומת אחרונה בשורת ההסטה ליצירת אלמנטים איכותיים

<u>*הוספת צומת ביניים*</u>



יש להוסיף צומת ביניים

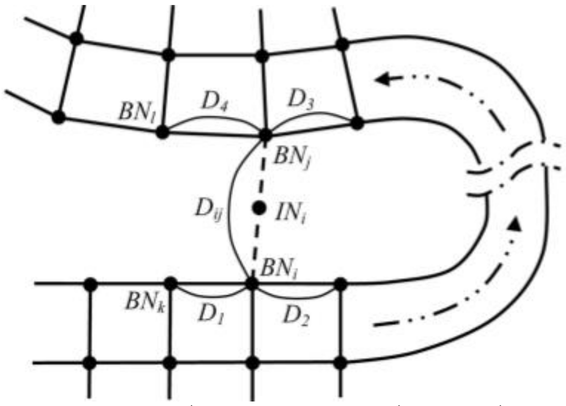
אם $D_{ij} > 1.453D_i$, עפ"י הנתון

איור 7: הוספת צומת ביניים - עפ"י המאמר

באיור 7. בהמשך, כאשר בודקים את D_k, נבדוק אם צריך להוסיף צומת ביניים בין IN_i ו-IN_k. אם כן, נוסף את האלמנט $(IN_{ij}, IN_i, IN_{ik}, BN_i)$.

אמיר ברדה
304848336

חלוקת השפות למניעת התנגשות בין אלמנטים:



איור 8: חלוקת התחום למניעת התנגשות בין שני אלמנטים

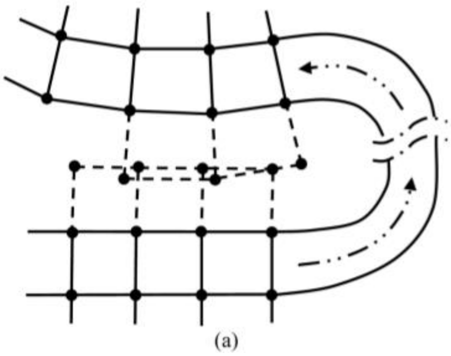
נשים לב שעל מנת שניתן יהיה לרשת עם אלמנטים מרובעים, כל לולאה סגורה צריכה להכיל מספר זוגי של צמתים (להוכיח). לכן, ייתכן שאין צורך להוסיף את הצומת $\boldsymbol{IN_i}$.

במאמר מצויין שצריך לבצע הפרדה אם יש חשש להתנגשות בין אלמנטים ש"פונים אחד לכיוון השני", אך לא הגדיר את התנאי הזה במפורש!!

כתוצאה מכך, הוגדר שאלמנטים "פונים אחד לכיוון של השני", כאשר הזווית בין הוקטורים הנורמלים שלהם היא $\boldsymbol{\pi - \alpha \le \theta \le \pi}$

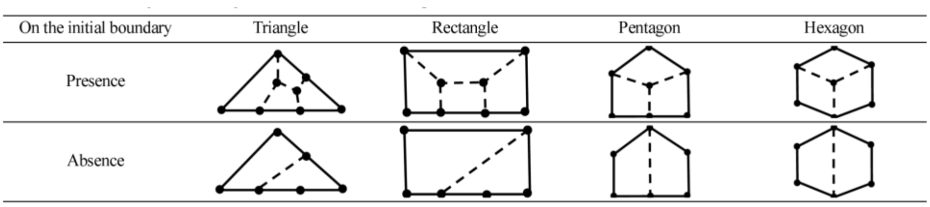
במהלך מימוש האלגוריתם נמצא שלערך $\boldsymbol{\alpha}$ חשיבות קריטית בקבלת רישות איכותי, והוא בייחוד מושפע מגודל האלמנטים האופייני שנבחר. כאשר הגודל האופייני קטן, $\boldsymbol{\alpha}$ צריך להיות קטן יותר, אחרת עלול להיווצר חלוקות רבות של התחום בקצוות וקבלת אלמנטים עדינים מדי. כאשר הגודל האופייני גדול, $\boldsymbol{\alpha}$ צריך להיות גדול יותר, אחרת עלול להיווצר מצב שלא תבוצע חלוקת תחום עבור אלמנטים שהם בסכנת התנגשות, ויווצרו התנגשויות.

שלב החלוקה מתאר את ההבדל העיקרי בין שיטת ה-**looping** לשיטת ה-**paving**. בניגוד לשיטת הריצוף (**paving**), האלגוריתם שמוצע במאמר אינו מרפא אלמנטים שהתנגשו ע"י תפירתם, אלא נמנע מהתנגשויות. האלגוריתם עובר על כל זוג צמתים בתחום, ובודק את המרחק ביניהם (D_{ij}), ראה איור **3**. במידה המרחק הנ"ל קטן מהסכום $(D_1 + D_2 + D_3 + D_4)/2$, שהוא בעצם סכום הממוצעים של D_1ו-D_2 ושל D_3ו-D_4. דוגמא לאיחוי אלמנטים לאחר התנגשות, בשיטת **paving**, נתונה איור 9.



איור 9: דוגמא לאיחוי אלמנטים בשיטת paving

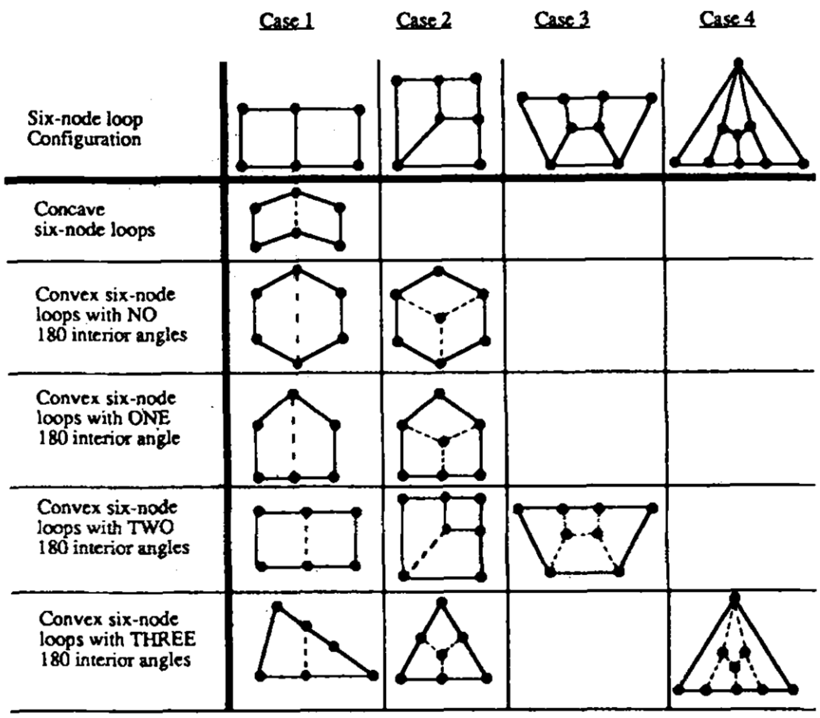
חלוקת מוגדרת של לאלמנטים עבור מעטפות בעלות שישה צמתים:



במידה ואין במעטפת אף צומת שהיא על המעטפת ההתחלתית, נחלק את האלמנט שרירותית לשני אלמנטים מרובעים.

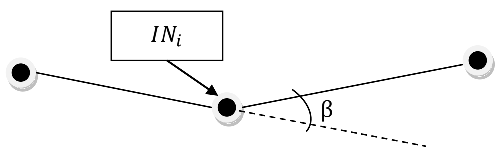
במידה וקיימת צומת במעטפת אזי נחלק עפ"י התבניות שנתונה במאמר (**talbert**)

אמיר ברדה
304848336



הפרמטרים שבהם ייעשה שימוש על מנת לבחור את חוצה האלמנטים הוא מספר הצמתים בעלי זווית **180°** והמספר המקסימלי של צמתים עוקבות כאלה.

במאמר נתון תצורה אידיאלית של צמתים לתיאור מצב של תחום בעל שישה צמתים. בפועל, התצורות שיתקבלו הן לרוב מקורבות לתצורה האידיאלית, ולכן יש להגדיר משתנה $\boldsymbol{\beta}$, כאשר צומת שיוצרת קו ישר מוגדרת ככזו שיוצרת סיבוב של $\boldsymbol{\beta}$ מעלות או פחות.



<u>החלקה מקומית</u>

לאחר יצירת שורת אלמנטים חדשה, השפה הקדמית החדשה נוטה להיות בעלת מראה מחוספס או "גס" (להראות תמונה), כתוצאה מפעולות כמו מחיקת והוספת צמתים, שתוארו ממקודם.
כאשר מבצעים החלקה מקומית, כל צומת מוזז בהתאם לצומת האב שלו, ושכניו בשפה הקדמית, ראה איור 10.

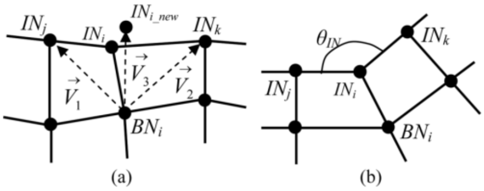


Fig. 8. Smoothing of a new internal boundary (loop), (a) Movement of a node, (b) no movement of a node.

איור 10: ביצוע החלקה מקומית לאחר הסטת השפה הקדמית פנימה

כפי שניתן לראות באיור 10, לאחר ההחלקה, מיקום הצומת המוזז הוא:

$$\vec{V_3} = \frac{\vec{V_1} + \vec{V_2}}{2}$$

במידה והזווית θ_{IN} מקיימת את קריטריון האיכות (1) $(45° \leq \theta_{IN} \leq 135°)$, אז לא תבוצע החלקה מקומית עבור אותו צומת, ראה איור 10.

<u>*החלקה גלובאלית*</u>

נעשה באמצעות השיטה הלפלסיאנית. (המאמר לא ציין שיש לבצע כמה מעברים על מנת להגיע להתכנסות!!)

עבור כל צומת שלא שלא על השפה:

$$(X_{node}, Y_{node}) = \frac{1}{N}\sum_{k=1}^{N}\left(X^k_{Connected}, Y^k_{Connected}\right)$$

כאשר (X_{node}, Y_{Node}) הוא המיקום שיש להזיז אליו את הצומת. N היא כמות הצמתים שמחוברים לצומת (המעלה של הצומת), ו-$\left(X^k_{Connected}, Y^k_{Connected}\right)$ היא הקואורדינטה של הצמתים המחוברות.

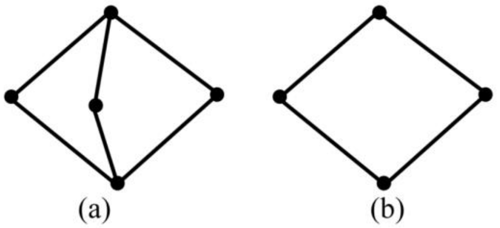
שיפור אלמנטים באיכות נמוכה

לאחר רישות והחלקה גלובלית, יש לעבור על כל אלמנט ברישות, ואם קיימת בו זווית פנימית גדולה מ-**135°**, לבדוק אם ניתן לבצע אחד מארבעת השיפורים הנ"ל:

מקרה 1: שני אלמנטים שחולקים שלושה צמתים

אם שני אלמנטים חולקים שלושה צמתים כפי שנראה באיור 11, בוודאות לאחד מהם זווית גדולה מ-180°, ולא נתין לבצע אנליזת **FEM**. הפיתרון הוא איחד שני האלמנטים.

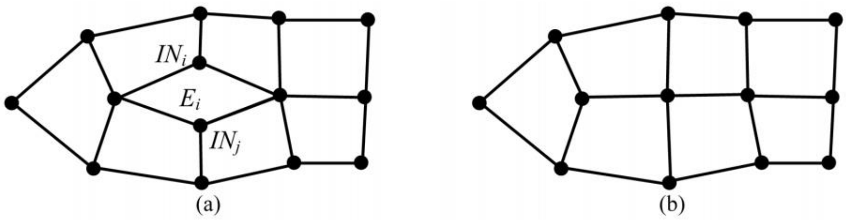
אמיר ברדה
304848336



איור 11: שני אלמנטים שחולקים שלושה צמתים

מקרה 2: אלמנט בעל שני צמתים נגדיות, כאשר כל צומת משותפת לשני אלמנטים.

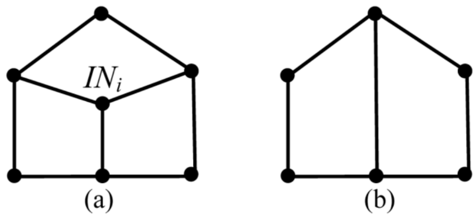
כפי שניתן לראות מאיור 12, לאלמנט E_i בד"כ תהיה זווית פנימית גדולה מ-$135°$, בשני הצמתים IN_i ו-IN_j. הפיתרון הוא מחיקת הצומת וריחד שתי הצמתים.



איור 12: אלמנט בעל שני צמתים נגדיות שמשותפות כל אחת לשני אלמנטים אחרים

מקרה 3: אלמנט בעל זווית גדולה בצומת שמשותפת לשני אלמנטים

אם לאלמנט יש זווית גדולה מ-$135°$ בצומת IN_i, שמשותפת לשני אלמנטים נוספים, כפי שנראה באיור 13. הפיתרון הוא מחיקת הצומת, לקבל שפה בעלת שישה צמתים, ושימוש בתבניות קבועות לחלוקת תחום בעל שישה צמתים.

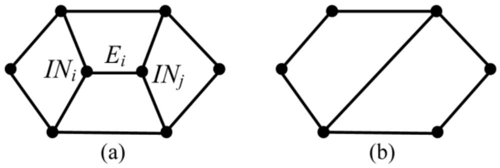


איור 13: אלמנט בעל זווית גדולה בצומת שמשותפת לשני אלמנטים נוספים

מקרה 4: ארבעה לאמנטים בתחום בעל שישה צמתים

אם שני צמתים (IN_i ו-IN_j) של אלמנט E משותפים כל אחד לשני אלמנטים נוספים, אזי יש ארבעה אלמנטים בתחום של שישה בעל שישה צמתים. לרוב לצמתים יצרו אלמנט בעל זווית גדולה מ-$135°$. הפיתרון הוא מחיקת הצמתים ויצירת תחום בעל שישה צמתים. לאחר מכן, יש לחלק את התחום הנ"ל בעזרת בתבניות קבועות לחלוקת תחום בעל שישה צמתים.

אמיר ברדה
304848336



איור 14: אלמנט בעל שני צמתים צמודים שכל אחד מהם משותף לשני אלמנטים

מימוש האלגורתים:

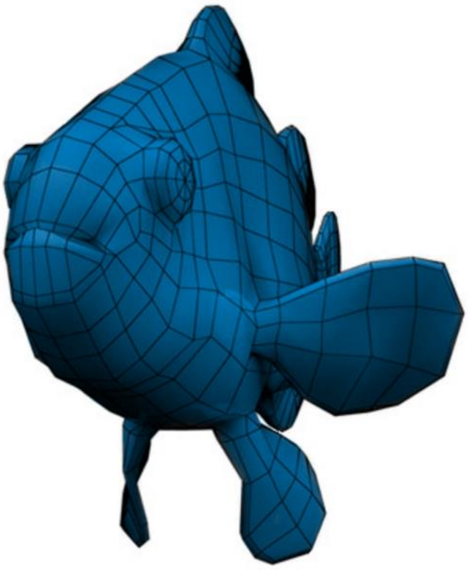
לצורך מימוש האלגוריתם הוגדר תור ***Q***, שמכיל מעטפות. בכל שלב מוציאים מעטפת מן התור, לצורך עיבוד, ו\או מוסיפים מעטפות חדשות לתור.

1. קליטת משטח יחיד (באיזה פורמט? איך מטפלים בקדחים?). **וגודל אופייני רצוי עבור אלמנט**
2. פיזור צמתים התחלתיים על עקומי **Bezier** שמתארים את מעטפת המשטח. הוסף את הצמתים במעטפת לתור המעטפות.
3. כל עוד תור המעטפות לא ריק:
4. בדוק אם במעטפת הקדמית בתור יש 6 צמתים או פחות. אם כן, השתמש בחלוקה מוגדרת מראש לאלמנטים והסר את המעטפת הקדמית מהתור. חוזר משלב 2.
5. השתמש בתנאי (??) על מנת לבדוק אם נדרשת חלוקה של התחום. אם כן, צור צומת אמצעית וחלק את התחום. הוסף את שתי המעטפות החדשות שנוצרו לסוף תור המעטפות, והסר את המעטפת הנוכחית (הקדמית בתור) מהתור. חזור משלב 2.
6. אם לא, הסט את הצמתים במעטפת הקדמית בתור, על מנת ליצור אלמנטים חדשים. השתמש ב(**משוואות**)**, לחישוב מיקום הצומת החדשה. הוסף את המעטפת החדשה שנוצרה לתור המעטפות.**
7. בצע החלקה מקומית למעטפת החדשה.
8. עבור על כל האלמנטים שלא על הגבול ובצע החלקה גלובלית.
9. עבור על כל האלמנטים ובצע מעבר לתיקון אלמנטים לא איכותיים.
10. בצע החלקה גלובלית שלא על הגבול ובצע החלקה גלובלית.

תוצאות:

האלגוריתם מומש ב-**C++**, תו שימוש בספריית **surface mesh** של **CGAL** כמבנה נתונים לבניית הרישות.
התוצאה הסופית היא תוכנית שמאפשרת קליטת קובץ **parasolid** שמכיל מידע לגבי משטח, ומרשתת את המשטח.

אמיר ברדה
304848336



איור 15: דוגמא לדג שרושת בעזרת surface mesh (מתוך האתר של CGAL)

אופטימיזציות אפשריות בקוד:

חישוב אפשרות להתנגשות ביו צמתים: עפ"י המאמר של **talbert**.

האלגוריתם נוסה על ארבעת הדגמים הבאים:

ביליוגרפיה:

1. https://www.rhino3d.com/nurbs
2. http://web.cs.wpi.edu/~matt/courses/cs563/talks/nurbs.html