

****

**عنوان:**

تولید شبکه دو بعدی بی‌سازمان مثلثی به روش Lawson

|  |  |  |
| --- | --- | --- |
| **نویسندگان** | مرتضی نامور |  |
| رضا ربیعی |  |
| **تاریخ تنظیم سند** | 1/2/1397 | |
| **شناسه سند** | **MC5F001F1** | |

**فهرست مطالب**

[فصل 1- راهنمای کاربری 1](#_Toc512359375)

[1-1- فایل ورودی 1](#_Toc512359376)

[1-2- اجرای برنامه 1](#_Toc512359377)

[1-3- فایل‌های خروجی 2](#_Toc512359378)

[1-4- توانایی‌ها و محدودیت‌ها 2](#_Toc512359379)

[فصل 2- اعتبارسنجی و نتایج 4](#_Toc512359380)

[2-1- تولید شبکه اولیه 4](#_Toc512359381)

[2-2- به روزرسانی (بازیابی) اضلاع مرزی شبکه 6](#_Toc512359382)

[2-3- حذف نقاط و المان‌های بیرون از میدان (مرز) 7](#_Toc512359383)

[2-4- ریز کردن شبکه 9](#_Toc512359384)

[فصل 3- تئوری و الگوریتم 12](#_Toc512359385)

[3-1- تولید مثلث بزرگ اولیه 13](#_Toc512359386)

[3-2- معرفی نقطه جدید و یافتن المانی که نقطه جدید درون آن قرار دارد 14](#_Toc512359387)

[3-3- اضافه کردن نقطه معرفی شده به شبکه و تشکیل المانهای جدید 14](#_Toc512359388)

[3-4- بررسی دلانی بودن المانهای همسایه 15](#_Toc512359389)

[3-5- دلانی کردن المانهای غیردلانی 16](#_Toc512359390)

[3-6- به روزرسانی اضلاع مرزی 17](#_Toc512359391)

[3-7- حذف نقاط و المان‌های خارج از میدان (مرز) 18](#_Toc512359392)

[3-8- ریز کردن شبکه با اضافه کردن نقطه 19](#_Toc512359393)

[3-8-1- تعیین تابع اندازه المان برای نقاط تشکیل دهنده شبکه درشت 20](#_Toc512359394)

[3-8-2- یافتن المانی که نقطه جدید درون آن قرار دارد 21](#_Toc512359395)

[3-8-3- تولید نقطه جدید 21](#_Toc512359396)

[3-8-4- اضافه کردن نقطه معرفی شده به شبکه و تشکیل المانهای جدید 21](#_Toc512359397)

[3-8-5- تهیه لیستی از مثلثهای جدید که باید دلانی بودن آنها بررسی گردد 21](#_Toc512359398)

[3-8-6- بررسی دلانی بودن آخرین مثلث موجود در لیست بالا 21](#_Toc512359399)

[3-8-7- دلانی کردن مثلثهای غیردلانی موجود در لیست 22](#_Toc512359400)

[3-9- تبدیل اطلاعات شبکه از حالت سلول محور به ضلع محور 22](#_Toc512359401)

[فصل 4- پیاده‌سازی و زیربرنامه‌های مورد استفاده 24](#_Toc512359402)

[فصل 5- مراجع 30](#_Toc512359403)

**چکیده:**

در این پروژه هدف تولید شبکه دلانی دو بعدی با استفاده از روش Lawson می­باشد. این روش یکی از روش های پایه­ای تولید شبکه دلانی می­باشد که تولید شبکه با استفاده از نقاط موجود در میدان انجام می شود. در این روش با معرفی نقطه جدید ابتدا مثلثی که این نقطه درون آن قرار دارد پیدا می شود و سپس با اتصال رئوس این مثلث به نقطه جدید معرفی شده، مثلث­های جدیدی بوجود می­آید که تنها دلانی بودن مثلث­های جدید بررسی شده و در صورت غیر دلانی بودن، دلانی می­شوند. در مقایسه با روش واتسن این روش دارای سرعت بسیار بالاتری بود و همچنین پیاده­سازی آن در دو بعد بمراتب ساده­تر است. این دو خصوصیت برتری چشمگیر این روش نسبت به روش واتسن را نشان می­دهد. این روش نیز مانند سایر روش­های تولید شبکه دلانی، تضمینی برای محافظت از مرزها ندارد که باید در پایان مراحل تولید شبکه، اضلاع مرزی بازیابی شوند، بدین صورت که اضلاع تمام منحنی های مرزی بررسی می شود و درصورتیکه یک ضلع مرزی در شبکه تولید شده موجود نباشد، یک نقطه اضافی در محل تقاطع این ضلع و اضلاع شبکه، تولید شده و با تولید مثلث های جدید، این نقطه به شبکه معرفی می شود. از نظر پیاده سازی این روش در مقایسه با سایر روش ها ساده بوده اما نقطه ضعف آن اینست که به ازای هر ضلع از دست رفته حداقل یک نقطه اضافی بر روی مرزها تولید می کند. همچنین در ادامه المان‌هایی که خارج از میدان (درون مرز داخلی یا بیرون مرز خارجی) قرار دارند، تعیین و به انتهای لیست المان ها منتقل می شوند و در انتها با تغییر تعداد نقاط و المان های شبکه، حذف این المان ها و نقاط تولید شده در روش های دلانی (برای تولید مثلث بزرگ اولیه) انجام می شود. روش ارائه شده در اینجا بسیار کارآمد بوده و گسترش آن برای شبکه های سه بعدی نیز امکانپذیر و از نظر برنامه نویسی ساده می باشد. استفاده از ساختار داده ای ضلع محور بدلیل کارآمدی آن و همچنین نیاز به حافظه کمتر در ذخیره سازی اطلاعات شبکه بطور گسترده ای مورد توجه قرار گرفته است. از آنجا که در فرایند تولید شبکه از ساختار داده ای سلول محورد استفاده می شود بنابراین لازم است در انتهای مراحل تولید شبکه ساختار داده ای تغییر یابد.

**کلمات کلیدی:** شبکه بی‌سازمان، دلانی، روش Lawson، بازیابی اضلاع مرزی، ریز کردن شبکه.

# راهنمای کاربری

## فایل ورودی

برای ورودی این زیر برنامه یک قالب مشخص، که سعی می شود در تمام برنامه های مربوط به حوزه تولید شبکه دو بعدی از آن استفاده گردد، بکار گرفته شده است. برای اطلاعات بیشتر در مورد این قالب و ساختار فایل ورودی می توانید به مستندات زیربرنامه Read\_2DMeshC مراجعه نمایید. تمام اطلاعات لازم باید در قالب یک فایل بنام MeshIn.cgid تهیه شده و در پوشه حاوی برنامه قرار داده شود.

## اجرای برنامه

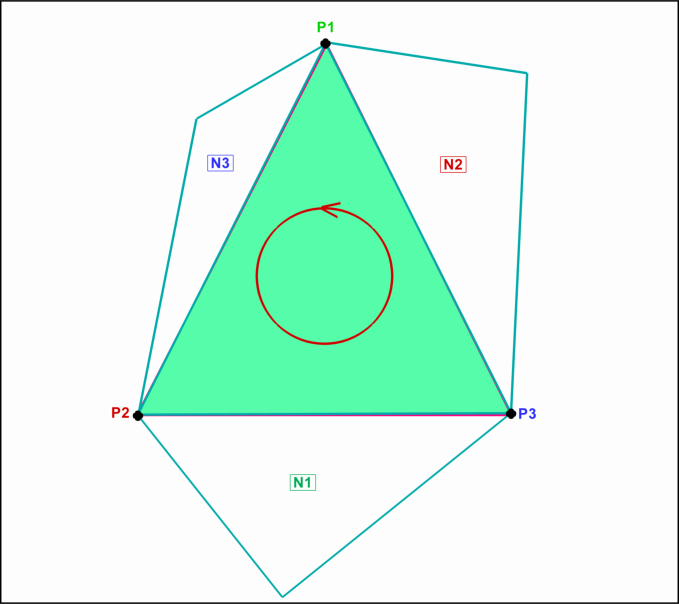
این برنامه تحت نرم افزار Comaq Visual Fortran توسعه یافته است بنابراین جهت اجرای آن بهتر است این نرم افزار را نصب نمایید. در اینجا یک پارامتری بنام Dim برای تعیین مقدار ماکزیمم بعد آرایه های استفاده شده، در نظر گرفته شده است. یک مقدار پیش فرض برای این پارامتر تعیین شده است و در صورتیکه با خطایArray bound exceed مواجه شدید باید مقدار پارامتر Dim را افزایش دهید. در صورتیکه برنامه با خطایStack Overflow مواجه شود، می توان با تغییر در تنطیمات نرم افزار Comaq Visual Fortran، Stack آن را افزایش داد. بهتر است در هنگام اجرای برنامه، نرم افزار Comaq Visual Fortran در حالت Debug قرار داشته باشد زیرا در حالت Release ممکن است با خطاهایی مواجه شوید.

## فایل‌های خروجی

یک قالب مشخص برای چاپ اطلاعات در نظر گرفته شده است تا با ساختار فایل های ورودی سازگاری داشته باشد. به این ترتیب ارتباط بین برنامه‌های نوشته شده براحتی امکانپذیر می باشد. برای اطلاعات بیشتر در مورد این قالب و ساختار فایل خروجی می‌توانید به مستندات زیربرنامه WriteMesh\_gid و Write2DMesh\_gid\_plt مراجعه نمایید. بنابراین اطلاعات خروجی با توجه به ساختارهای در نظر گرفته شده در فایل­هایی با فرمت Plt و Gid چاپ می‌شوند. همچنین اطلاعات مرزهای شبکه توسط زیربرنامه WriteBoundCrv\_cgid\_plt و در فرمت Plt ذخیره می‌شوند.

## توانایی‌ها و محدودیت‌ها

این برنامه شبکه‌ای را از ورودی می‌خواند که اطلاعات آن بصورت سلول محور[[1]](#footnote-1) ذخیره شده باشد، پس باید در نظر داشت که برای یک شبکه دوبعدی سلول محور نحوه ذخیره نقاط تشکیل دهنده هر مثلث، بگونه ایست که بنابر قانون دست راست جهت بردار عمود بر مثلث بطرف خارج از صفحه و برای یک شبکه سطحی این جهت بسمت داخل جسم می باشد. همچنین نحوه ذخیره همسایه ها متناظر با ترتیب نقاط تشکیل دهنده المان می باشد. برای مثال اولین همسایه مثلث زیر، روبری اولین نقطه تشکیل دهنده آن است و به همین ترتیب سایر همسایه ها ذخیره شده اند. برای روشن شدن این موضوع به ‏شکل (1) توجه کنید.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **اولین نقطه** | **دومین نقطه** | **سومین نقطه** | **اولین همسایه** | **دومین همسایه** | **سومین همسایه** |
| P1 | P2 | P3 | N1 | N2 | N3 |

1. نحوه ذخیره نقاط تشکیل دهنده و همسایه‌های یک المان مثلثی

با توجه به اینکه در روش‌های دلانی تضمینی برای نگه‌داری مرزها نمی‌باشد، بنابراین لازم است در انتهای مراحل مثلث‌بندی، مرزهای از دست رفته بازیابی شود. همچنین حذف المان‌هایی که خارج از دامنه (خارج از مرز بیرونی و درون مرزهای داخلی) تولید شده‌اند، انجام می‌پذیرد. از آنجا که ابتدا یک شبکه درشت تولید می‌شود، با استفاده از یک تابع اندازه، المان‌های شبکه ریز می‌شوند و در انتها اطلاعات شبکه بصورت ضلع محور[[2]](#footnote-2) ذخیره می‌شود.

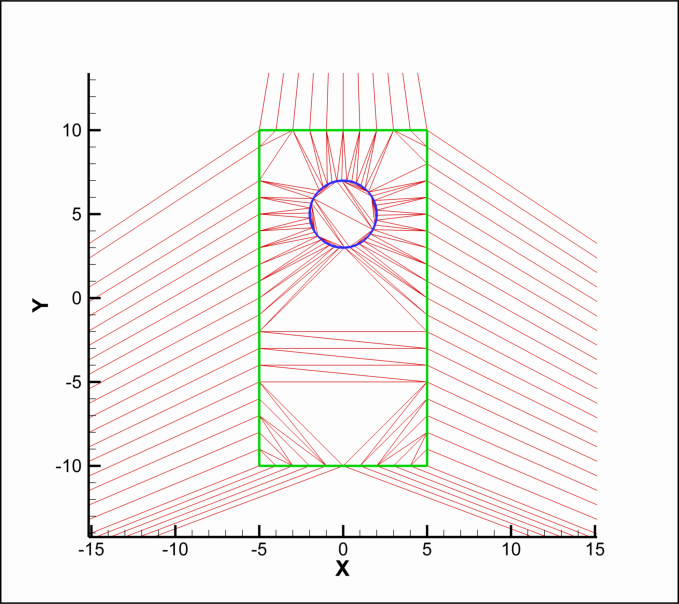
لازم است یادآوری شود که در اینجا از ابزارهای افزایش کیفیت شبکه استفاده نشده است، بنابراین نمی‌توان انتظار داشت که یک شبکه هموار و با کیفیت بالا تولید شده باشد. برای اینکار باید از روش های افزایش کیفیت برای بالا بردن کیفیت شبکه‌های تولید شده، استفاده گردد. همچنین این برنامه فقط برای المان‌های مثلثی شکل دو بعدی کاربرد دارد.

# اعتبارسنجی و نتایج

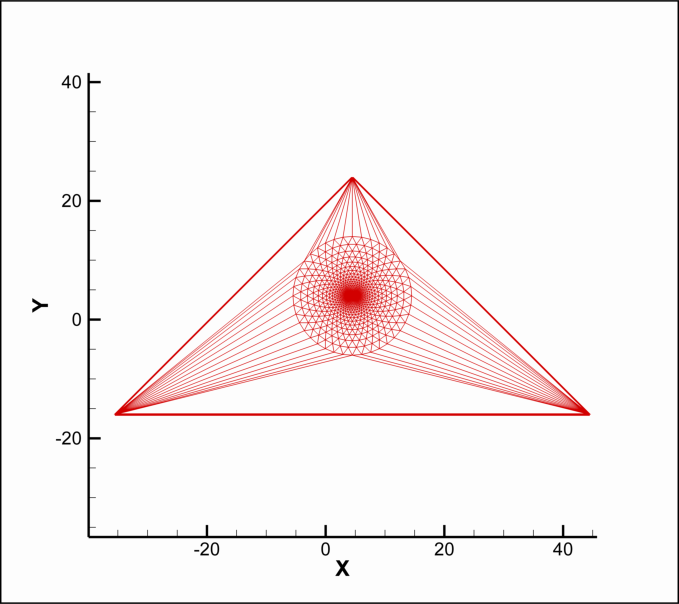
## تولید شبکه اولیه

در این بخش چند نمونه از شبکه­های تولید شده توسط برنامه آورده شده است. اولین شبکه مربوط به میدان قرار گرفته شده در بین یک دایره و مستطیل اطراف آن می باشد. هرچند این یک هندسه ساده می باشد اما وجود زوایای افقی و عمودی در اضلاع مرزی این هندسه قابل توجه می باشد. همچنین وجود لبه­های تیز که در بیشتر کدهای تولید شبکه ایجاد مشکل می­کند در این هندسه وجود دارد. همانگونه که از شبکه نشان داده شده مشخص است، تنها از نقاط مرزی برای تولید شبکه استفاده شده است. اینگونه شبکه ها بعنوان شبکه درشت یا اولیه مطرح می باشد که در بسیاری از موارد از آنها استفاده می شود. شبکه دوم مرزهای ایرفویل NACA0012 را نشان می­دهد. همانگونه که مشخص است یک شبکه مثلثی با کیفیت در اطراف آن تولید شده است. لازم بذکر است که نقاط غیر مرزی این شبکه از یک شبکه باسازمان گرفته شده است. در هندسه چهارم که یک ایرفویل سه المانه را نشان می­دهد، علاوه بر گسستگی­هایی که در مزر داخلی وجود دارد وجود نواحی مقعر در مرز داخلی می­تواند مشکل ساز باشد. در اینجا نیز نقاط غیر مرزی از یک روش تولید شبکه که در آن نقاط غیر مرزی بصورت خودکار تولید شده است استفاده گردیده.

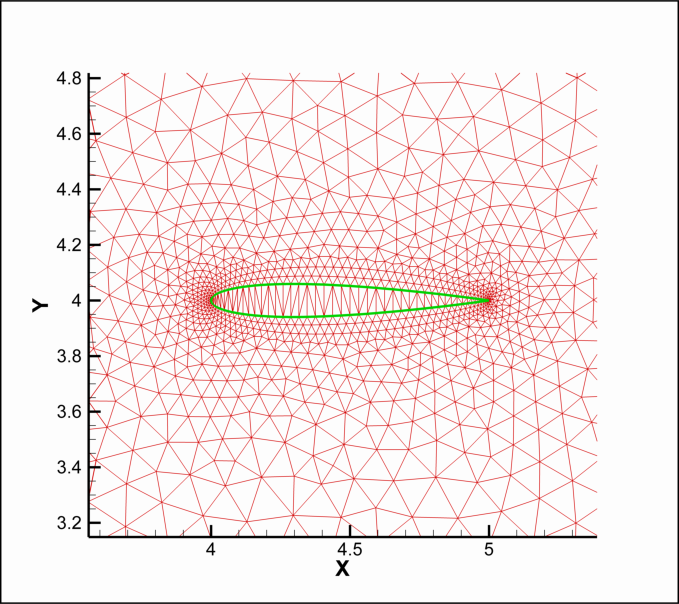
نکته قابل توجه در این هندسه ها اینست که مرزهای جسم در فرایند تولید شبکه از بین نرفته است هرچند بازیابی مرزها در این برنامه انجام نشده است. همانگونه که مشاهده می شود در این شبکه ها مثلث های تولید شده در خارج از دامنه حذف نشده است.



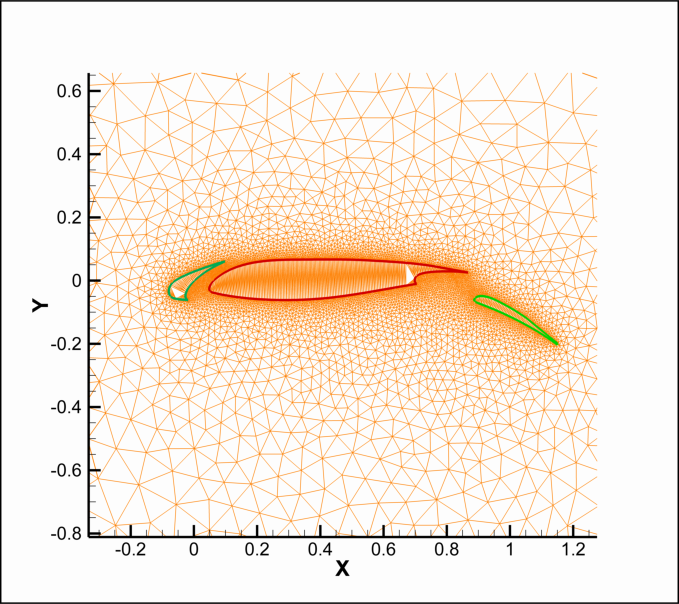
1. تولید شبکه در میدان بین یک دایره و مستطیل



1. تولید شبکه در اطراف ایرفویل Naca0012 (نمای دور)



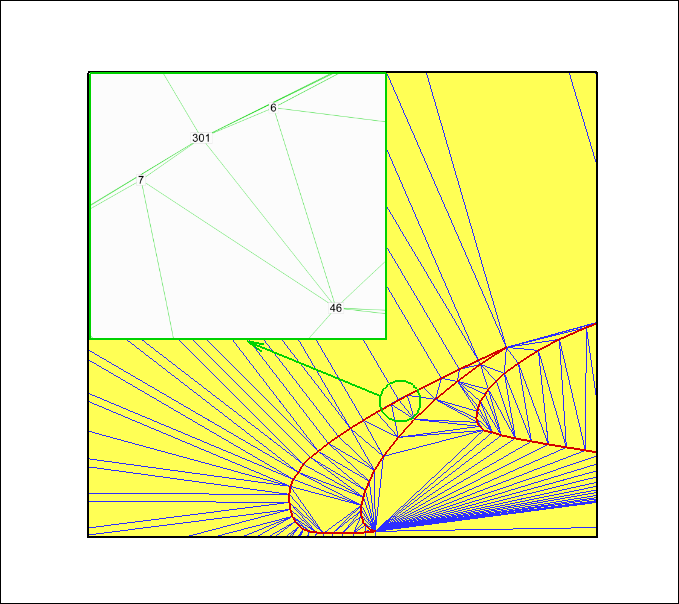
1. تولید شبکه در اطراف ایرفویل Naca0012 (نمای نزدیک)



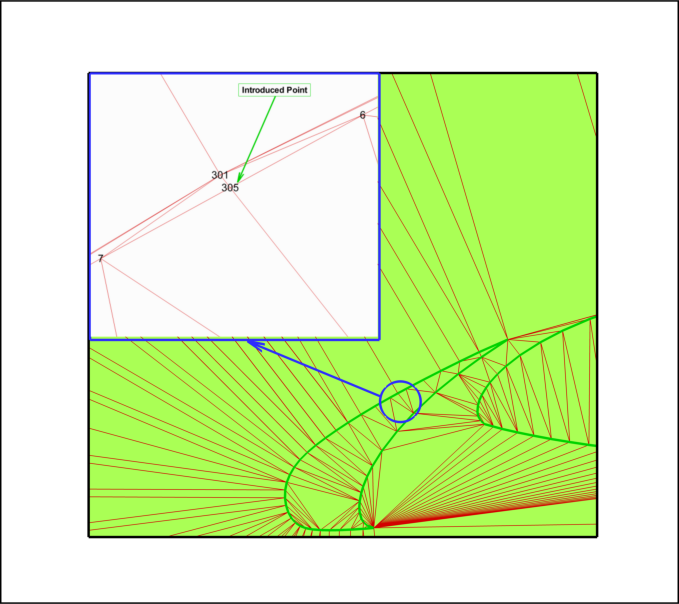
1. تولید شبکه در اطراف ایرفویل سه المانه

## به روزرسانی (بازیابی) اضلاع مرزی شبکه

در این بخش یک نمونه شبکه­ که در آنها اضلاع مرزی بازیابی نشده، آورده شده است. همانگونه که در ‏شکل (6) مشاهده می شود ضلع 6-7 که جزء اضلاع مرزی می باشد در فرایند تولید شبکه (با استفاده از نقاط مرزی) تشکیل نشده است. جهت بازیابی این ضلع مرزی یک نقطه به محل نقاطع ضلع مرزی 6-7 و اضلاع مثلث های موجود در میدان (مثلثی که یک نقطه آن شامل نقطه ابتدایی ضلع 6-7 است) تولید می شود. سپس با تولید مثلث های جدید این نقطه به شبکه اضافه شده و در نتیجه ضلع مرزی بازیانی نشده که اکنون به دو قسمت تبدیل شده است، بازیابی می گردد(‏شکل (7)). لازم است اشاره شود که به دلیل اینکه نقطه جدید بر روی مرزها تولید شده، باید ضلع مرزی اولیه (برای مثال ضلع 6-7) به دو بخش تبدیل شود. بنابراین در اینجا اطلاعات مرزها نیز بروزرسانی می شود.



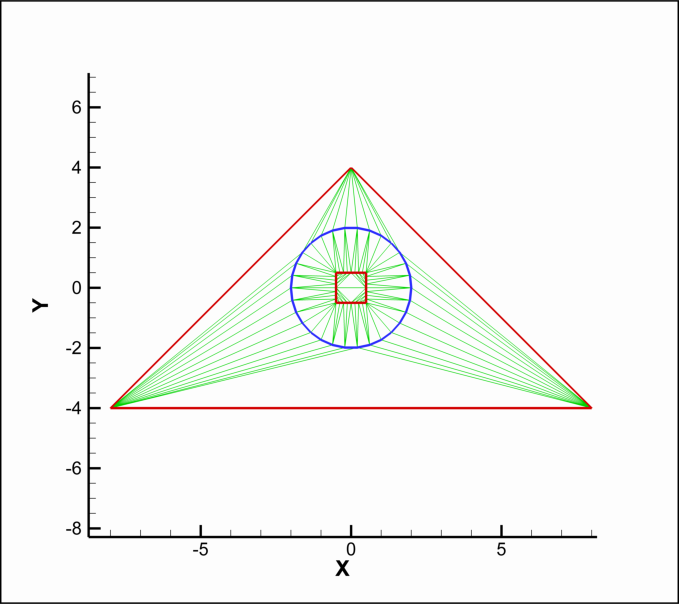
1. ضلع بازیابی نشده در فرآیند تولید شبکه دلانی



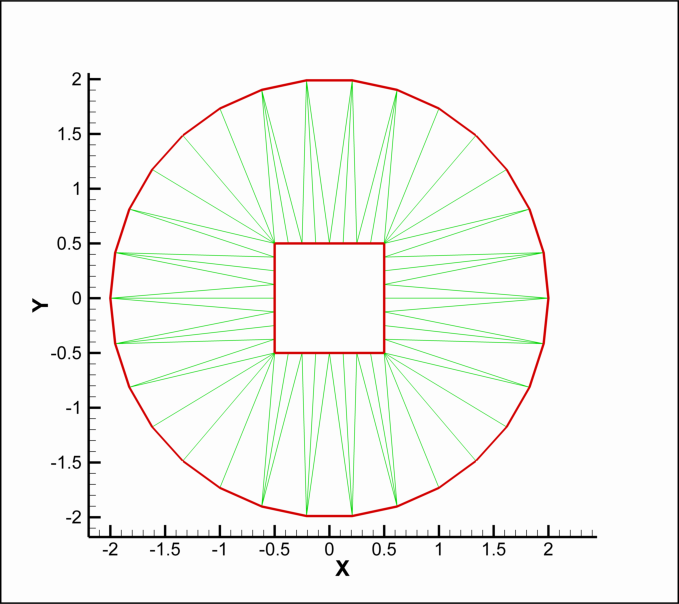
1. بازیابی ضلع از دست رفته با اضافه کردن نقطه

## حذف نقاط و المان‌های بیرون از میدان (مرز)

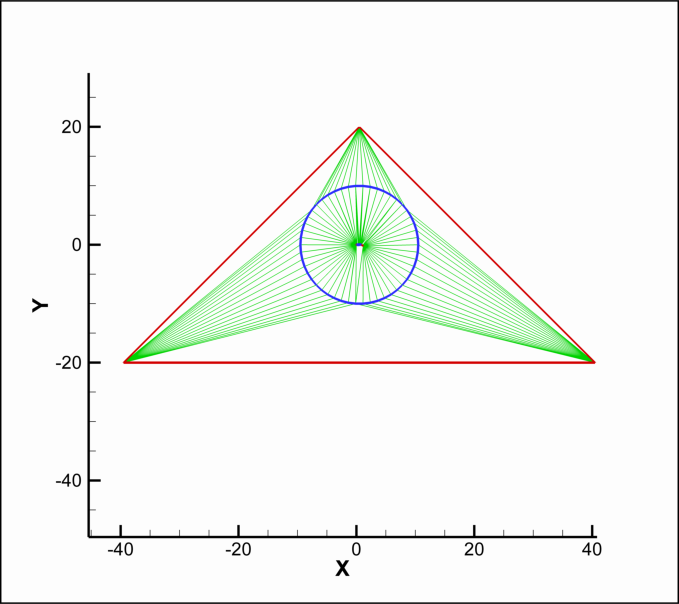
در این بخش چند نمونه از شبکه­هایی که در آنها نقاط و المان های خارج از میدان حذف نشده اند برای اعتبارسنجی برنامه حاضر مورد استفاده قرار گرفته است. اولین شبکه مربوط به میدان قرار گرفته شده در بین یک دایره و مربع اطراف آن می باشد. هرچند این یک هندسه ساده می باشد اما وجود زوایای افقی و عمودی در اضلاع مرزی این هندسه قابل توجه می باشد که ممکن است در هنگام پیدا کردن تقاطع خطوط برنامه دچار مشکل شود. در هندسه دوم که یک ایرفویل سه المانه را نشان می­دهد، علاوه بر گسستگی­هایی که در مزر داخلی وجود دارد وجود نواحی مقعر در مرز داخلی می­تواند مشکل ساز باشد. در هندسه سوم که دارای چندین زائده در مرز داخلی می باشد، برنامه تدوین شده بخوبی توانسه است المان های ناخواسته را حذف نماید.



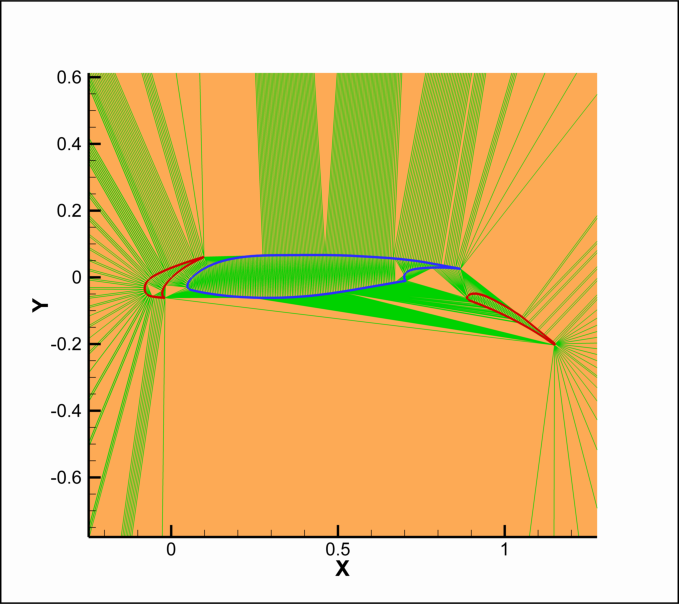
1. شبکه اولیه تولید شده در اطراف یک مربع (نمای دور)



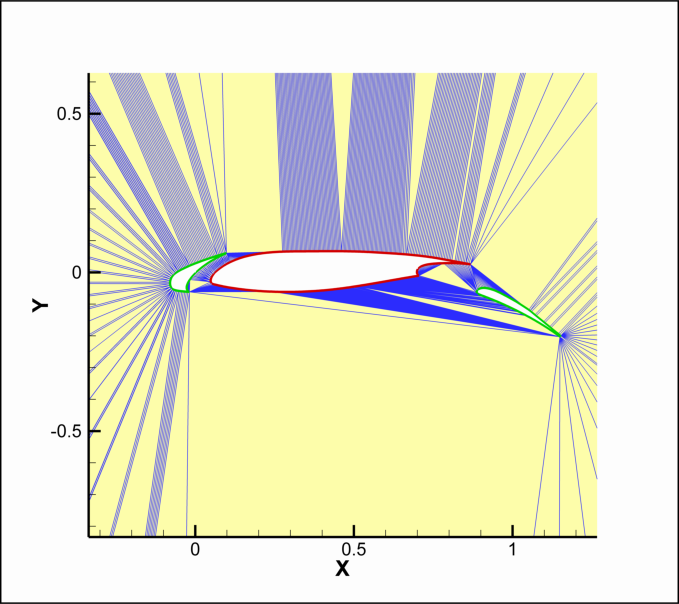
1. حذف المان ها در شبکه تولید شده در اطراف یک مربع



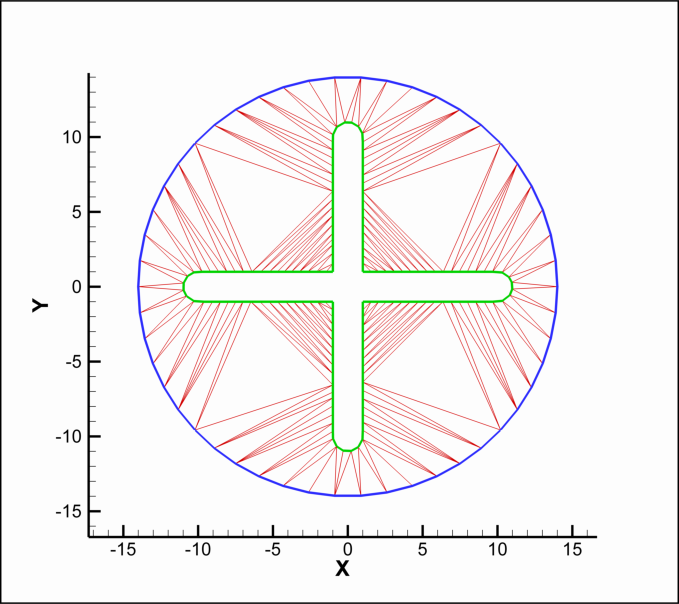
1. شبکه اولیه تولید شده در اطراف ایرفویل سه المانه (نمای دور)



1. شبکه اولیه تولید شده در اطراف ایرفویل سه المانه (نمای نزدیک)



1. حذف المان ها در شبکه تولید شده در اطراف ایرفویل سه المانه

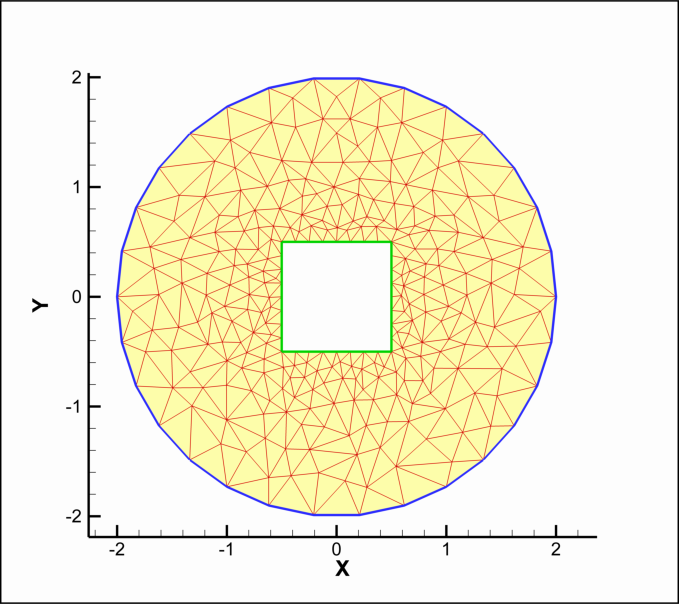


1. حذف المان ها در شبکه تولید شده در اطراف یک پره دوار

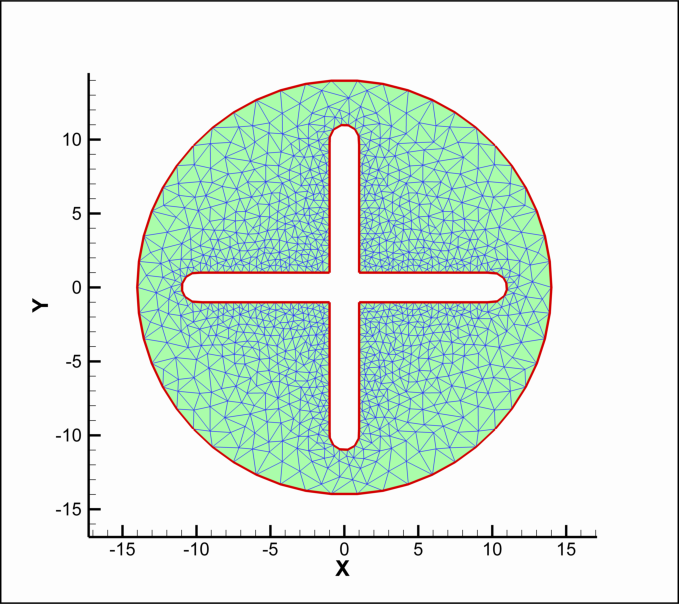
## ریز کردن شبکه

در این بخش چند نمونه از شبکه­های تولید شده توسط برنامه آورده شده است. لازم است یادآوری شود که در اینجا از ابزارهای افزایش کیفیت استفاده نشده است بنابراین نمی توان انتظار داشت که یک شبکه هموار و با کیفیت مناسب تولید شده باشد. برای اینکار باید از روش های افزایش کیفیت برای بالا بردن کیفیت شبکه های تولید شده، استفاده گردد.

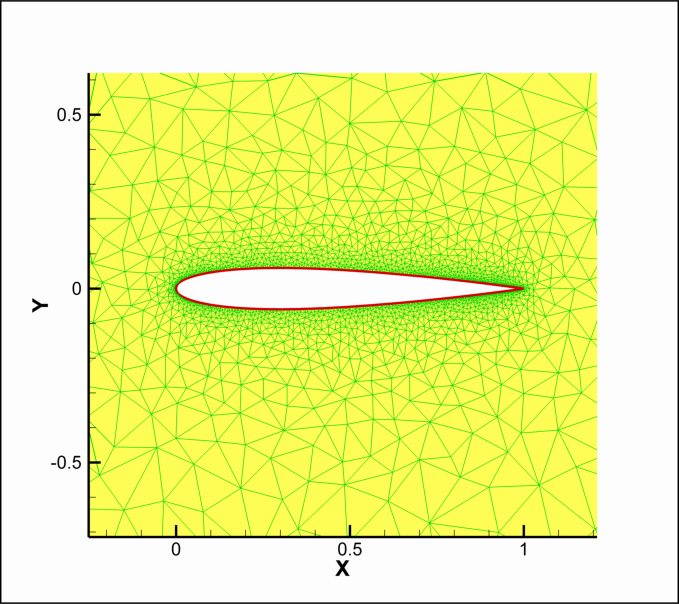
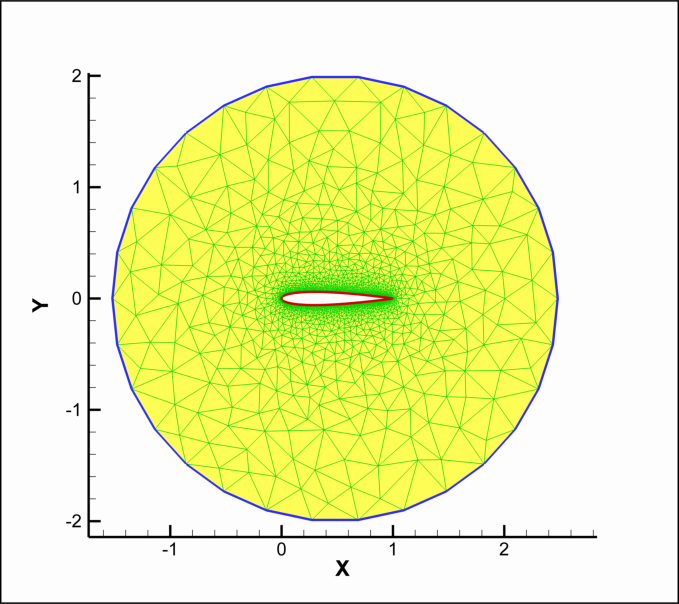
اولین شبکه مربوط به میدان اطراف یک مربع می­باشد که یک هندسه ساده برای تولید شبکه است. دومین شبکه مربوط به یک پره می باشد. در شبکه سوم مرزهای یک هندسه هوافضایی را نشان می­دهد. در هندسه چهارم که یک ایرفویل دو المانه را نشان می­دهد، علاوه بر گسستگی­هایی که در مزر داخلی وجود دارد وجود، نواحی مقعر در مرز داخلی قابل توجه می باشد. اجرای این برنامه بر روی این هندسه نسبتا پیچیده نشان از توانایی برنامه کامپیوتری ارائه شده دارد.



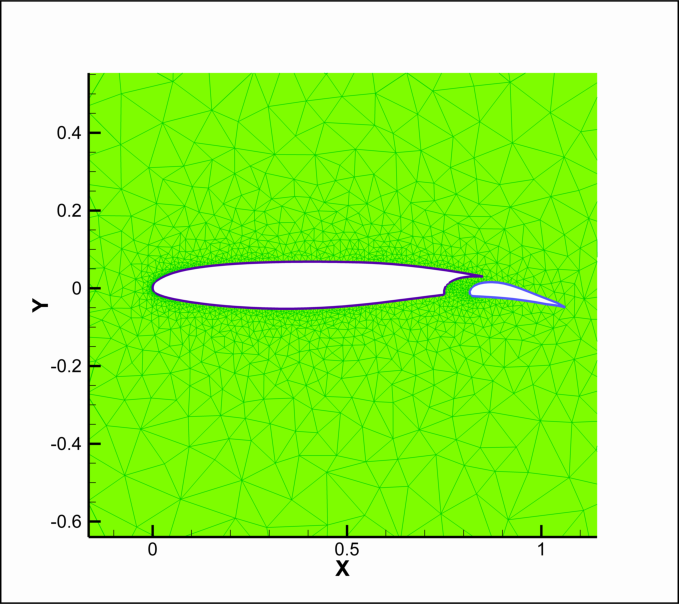
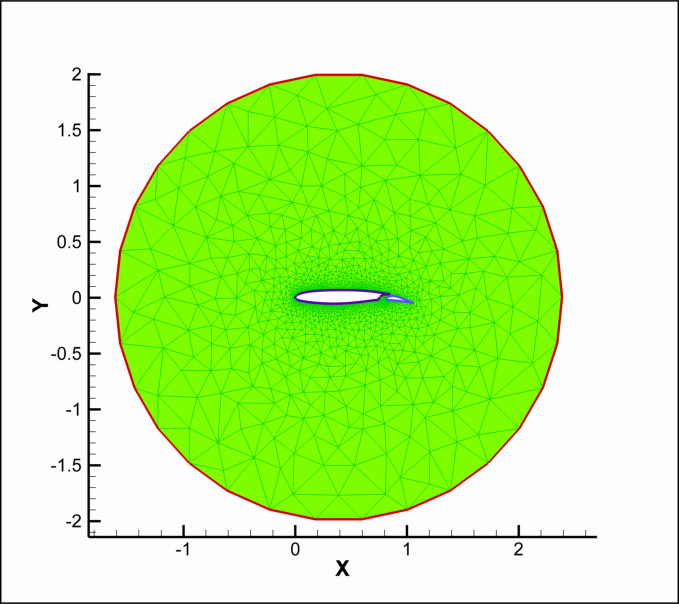
1. تولید شبکه در اطراف یک مربع



1. تولید شبکه در اطراف یک پره



1. تولید شبکه در اطراف ایرفویل NACA0012



1. تولید شبکه در اطراف ایرفویل دو المانه

# تئوری و الگوریتم

در این بخش تئوری و الگوریتم های روش تولید شبکه لاسن بررسی شده و در مراحل بعدی نحوه پیاده سازی آنها بطور مفصل آورده می شود.

روش لاسن[[3]](#footnote-3) یکی از روش­های مطرح در تولید شبکه مثلثی دوبعدی دلانی می­باشد که بعلت سادگی در پیاده­سازی بعنوان یک روش پایه برای توسعه شبکه سه بعدی مورد توجه قرار گرفته است. این روش اولین بار توسط لاسن معرفی گردید[1]. یکی از ویژگی­های بارز این روش سرعت بالای آن در مقایسه با سایر روش­های تولید شبکه دلانی می­باشد. سرعت این روش از مرتبه n.Log(n) می­باشد که در آن n نشان­دهنده تعداد مثلث­ها می­باشد. برخی از مقالاتی که در زمینه تولید شبکه سه بعدی منتشر شده­اند از این روش استفاده نموده­اند[2] و نشان داده‌اند که نحوه پیاده­سازی آن در دو بعد بگونه­ای بوده که توسعه این روش به سه بعد براحتی انجام شود.

در این تحقیق تنها تولید شبکه دو بعدی مورد نظر می­باشد اما نحوه پیاده­سازی آن با در نظر گرفتن برخی از مسائل مربوط به تولید شبکه سه بعدی انجام شده است. در این روش با معرفی نقطه جدید به شبکه موجود، تنها المان‌های جدید تولید شده از نظر ارضای خاصیت دلانی بررسی می‌شوند. مراحل تولید شبکه بی­سازمان دلانی، با استفاده از این روش که در گزارش حاضر پیاده­سازی شده است، بشرح زیر می­باشد:

1. تولید مثلث بزرگ اولیه.
2. یافتن المانی که نقطه جدید درون آن قرار دارد.
3. اضافه کردن نقطه معرفی شده به شبکه و تشکیل المان‌های جدید.
4. بررسی دلانی بودن المان‌های همسایه.
5. دلانی کردن المان‌­های غیردلانی.

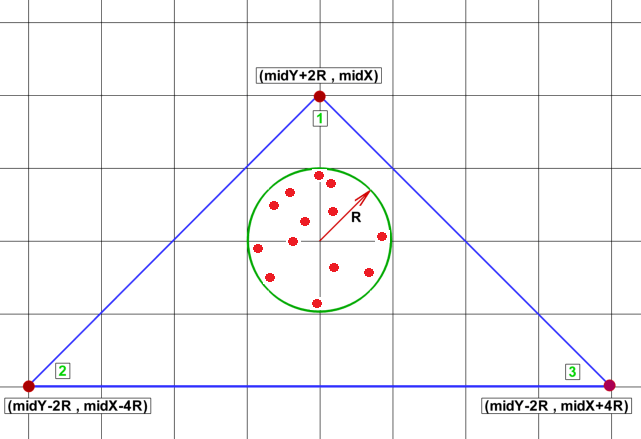
رجوع به مرحله اول و انجام هر یک از گام­های تعریف شده و انجام مثلث­بندی.

1. به روزرسانی (بازیابی) اضلاع مرزی.
2. حذف نقاط و المان‌های خارج از میدان (مرز).
3. ریز کردن شبکه با اضافه کردن نقطه.
4. تبدیل اطلاعات سلول محور به ضلع محور.

در ادامه هر کدام از مراحل فوق بطور مفصل توضیح داده می شود.

## تولید مثلث بزرگ اولیه

هر کدام از نقاط موجود باید به یک شبکه دلانی معتبر معرفی گردد. بنابراین باید همیشه یک شبکه دلانی معتبر وجود داشته باشد. به این دلیل لازم است قبل از معرفی اولین نقطه، یک شبکه وجود داشته باشد. گاهی اوقات از دو مثلث بزرگ که تشکیل یک شبکه می­دهند بعنوان شبکه اولیه استفاده می­شود اما ساده ترین شبکه، استفاده از یک مثلث بزرگ می­باشد که تمام نقاط موجود را در خود جای دهد. این المان اولیه می­تواند به اندازه دلخواه بزرگ باشد اما باید توجه داشت که اندازه بیش از حد آن نسبت به ابعاد شبکه مورد نظر، بر روی دقت محاسبات تاثیر می­گذارد. بنابراین لازم است اندازه آن فقط تا حدی بزرگ باشد که اطمینان حاصل شود تمام نقاط موجود را در بر می گیرد.



1. تشکیل مثلث بزرگ اولیه

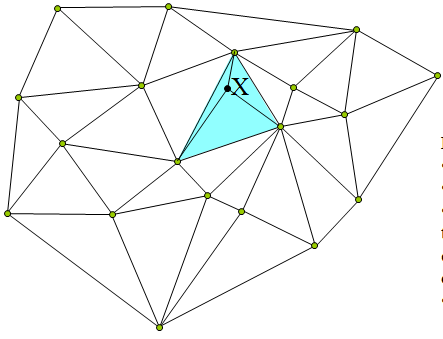
## معرفی نقطه جدید و یافتن المانی که نقطه جدید درون آن قرار دارد

پس از معرفی نقطه جدید باید المانی که نقطه درون آن قرار دارد، پیدا شود. برای این منظور معمولا از یک Walk search استفاده می­گردد تا سرعت تولید شبکه بالا باشد. اما در پروژه حاضر بدلیل اینکه هدف تولید یک شبکه اولیه می­باشد این نوع جستجو انجام نشده است و بجای آن تمام المان­های موجود از نظر قرار گرفتن نقطه جدید درون آنها بررسی می­گردد.

جهت یافتن المان کاندید (المانی که نقطه جدید درون آن قرار دارد) باید موقعیت نقطه جدید نسبت به اضلاع یک المان تعیین شود. اگر یک نقطه در سمت چپ (با توجه به جهت ذخیره رئوس مثلث که باید بطور پادساعتگرد باشد) هر سه ضلع یک المان باشد، نقطه درون المان قرار دارد.

## اضافه کردن نقطه معرفی شده به شبکه و تشکیل المان­های جدید

اضافه کردن نقطه جدید به شبکه بطور ساده با متصل کردن نقطه جدید به رئوس المانی که نقطه جدید درون آن قرار دارد، انجام می گیرد.



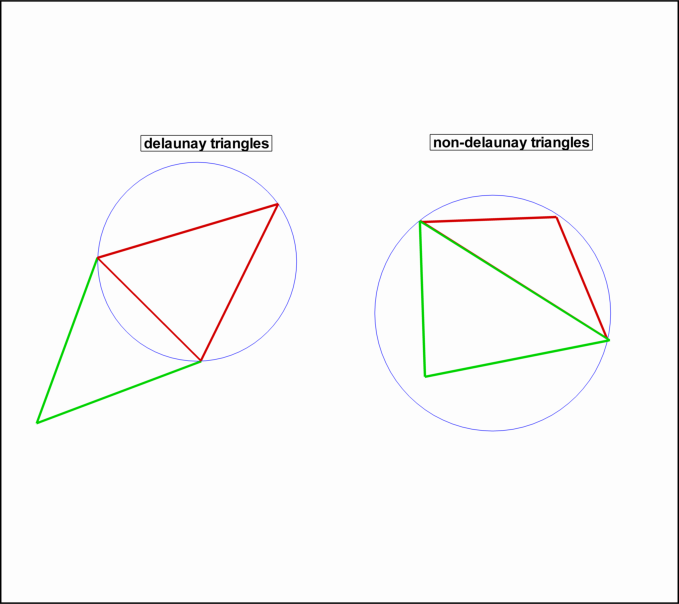
1. اضافه کردن نقطه به شبکه

در اینجا لازم است توجه شود که ترتیب ذخیره نقاط یک مثلث باید بصورت پادساعتگرد باشد. همچنین باید نحوه ذخیره همسایه های بگونه ای باشد که شماره هر همسایه متناظر با راس غیر مشترک باشد. برای روشن شدن این موضوع به ‏شکل (1) توجه کنید.

## بررسی دلانی بودن المان­های همسایه

پس از تشکیل المان­های جدید لازم است بررسی شود که آیا المان های جدید دلانی می­باشند یا خیر. بنابراین لیستی از المان­های جدید تشکیل شده و همسایه آنها تهیه می­گردد و دلانی بودن آنها مورد بررسی قرار می­گیرد.

یکی از مهمترین بخش­های این روش تولید شبکه، تشخیص دلانی یا غیر دلانی بودن دو المان همسایه می­باشد. برای این منظور دو المان زیر را در نظر بگیرید. در صورتیکه نقطه P4 درون دایره محیطی مثلث تشکیل شده از نقاط P1/P2/P3 قرار داشته باشد، این دو المان دلانی نیستند. در این حالت با جابجایی ضلع مشترک، این دو المان دلانی می­شوند. همچنین در صورتیکه نقطه P4 بر روی دایره محیطی قرار داشته باشد، این دو المان دلانی می­باشند ولی یکتا نیستند.



1. تشخیص دلانی یا غیر دلانی بودن دو المان در دوبعد

برای پیاده­سازی این روش، دترمینان این چهار نقطه محاسبه می­گردد و در صورتیکه مقدار دترمینان:

1. بزرگتر یا مساوی صفر باشد آنگاه دو المان دلانی هستند.
2. کوچکتر از صفر باشد آنگاه دو المان غیردلانی هستند.

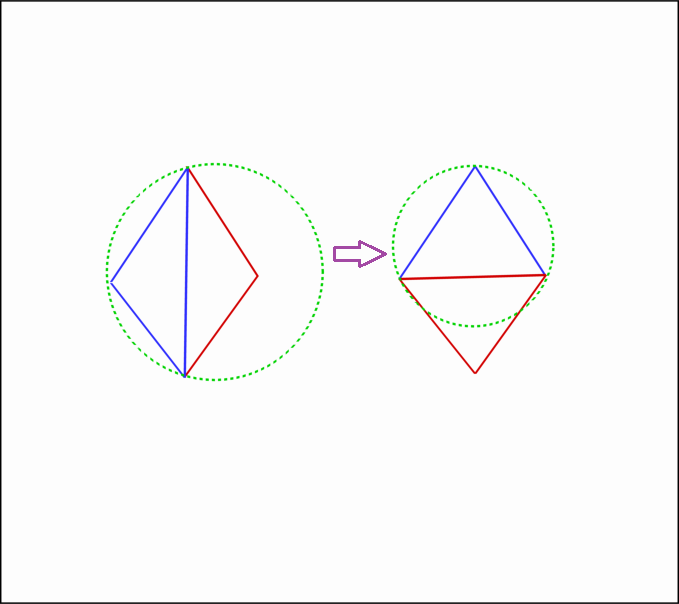
دترمینان فوق بصورت زیر است:

|  |  |
| --- | --- |
|  |  |

در اینجا لازم است نقطه چهارم متعلق یه مثلث همسایه و سه نقطه دیگر متعلق به مثلث اصلی باشد. همچنین باید ترتیب نقاط مثلث اصلی در خلاف جهت عقربه های ساعت باشد.

## دلانی کردن المان­های غیردلانی

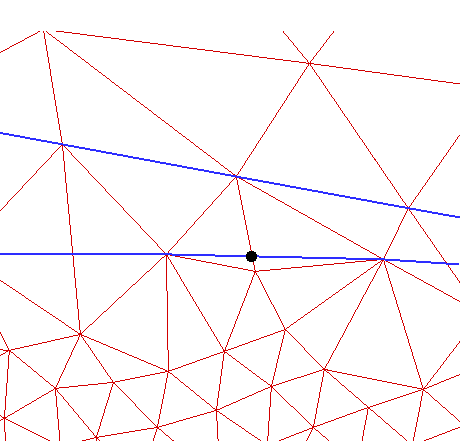
در فضای دو بعدی با جابجایی ضلع مشترک دو المان غیردلانی همسایه، این دو المان دلانی می شوند. برای مثال شکل زیر نحوه انجام اینکار را نشان می دهند.



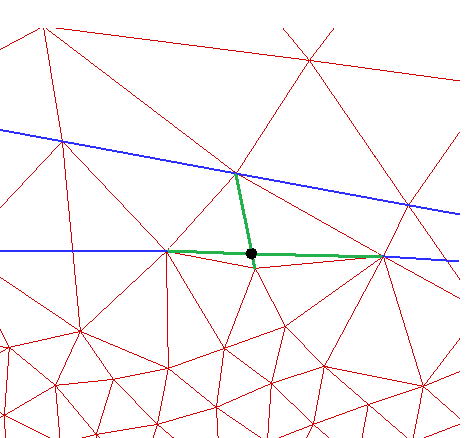
1. دو مثلث غیردلانی قبل از جابجایی ضلع مشترک

## به روزرسانی اضلاع مرزی

همانگونه که قبلا نیز اشاره شده روش دلانی تضمینی برای محافظت از مرز ندارد. بنابراین پس از تولید شبکه در کل میدان با استفاده از تمام نقاط موجود، باید از تشکیل اظلاع مرزی اطمینان حاصل شود. به این منظور پس از ایجاد مثلث بندی دلانی اضلاع مرزی بررسی شده و در صورت از بین رفتن، با تولید نقاط جدید، اضلاع مرزی بازیابی می شوند. برای اینکار ابتدا نقطه تقاطع ضلع مرزی و مثلث ایجاد شده در میدان محاسبه می­شود سپس یک نقطه جدید در نقطه تقاطع ایجاد می­شود و یک مثلث­بندی جدید با استفاده از دو مثلث منسوب به این ضلع ایجاد می­شود.



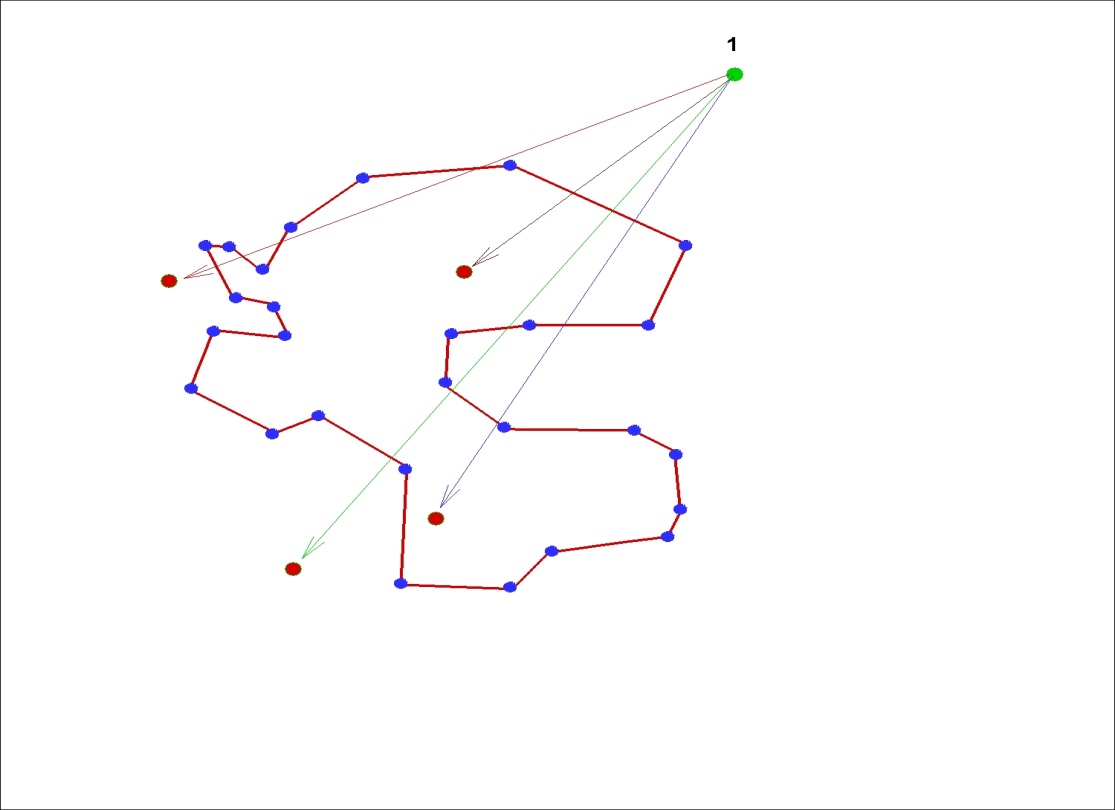
1. از بین رفتن اضلاع مرزی



1. بازیابی اضلاع مرزی با ایجاد مثلث بندی جدید

## حذف نقاط و المان‌های خارج از میدان (مرز)

پس از تولید شبکه در کل میدان با استفاده از تمام نقاط مرزی باید المان­هایی که خارج از مرز خارجی یا درون مرز داخلی قرار دارد، حذف گردند. جهت حذف المان­های موجود در ناحیه غیردلخواه از این خاصیت استفاده می­شود که برای ورود به یک محیط بسته، از یک نقطه خارج از آن، یک بار باید از مرزها عبور کنیم و جهت خروج از آن، بعد از ورود به آن، دو بار باید از مرزها عبور کنیم. به همین ترتیب اگر بخوایم از یک نقطه خارج از محیط بسته به آن وارد شویم باید به تعداد یک عدد فرد از مرزها عبور کنیم و جهت خروج از آن باید به تعداد عدد زوج از مرزها عبور کنیم. برای مثال شکل زیر را در نظر بگیرید که قرار است از نقطه 1 به نقاط دیگر برویم. مشاهده می­شود که اگر نقطه­ای که باید به آن برسیم درون ناحیه دلخواه باشد به تعداد یک عدد فرد مرزها قطع می­شوند و در صورتیکه نقطه در خارج از ناحیه دلخواه قرار داشته باشد، به تعداد یک عدد زوج مرزها قطع می­شود.



1. عملیات حذف نقاط و المان‌های خارج از میدان

لازم بذکر است که در هیچکدام از مقالات موجود در زمینه تولید شبکه چنین الگوریتمی ارائه نشده است و این کار یکی از نکات بسیار مهم این گزارش است که پیاده­سازی آن نیز بسیار ساده است و از نظر عددی بسیار قوی می باشد.

## ریز کردن شبکه با اضافه کردن نقطه

ریز کردن شبکه یا تولید نقاط جدید به روش‌های مختلفی همانند مراجع [3] و [4] انجام می‌شود. در روش حاضر المانی که قابلیت دریافت نقطه جدید را دارد با اضافه کردن یک نقطه در مرکز آن، المان مورد نظر ریز می شود. در اینجا المانی قابلیت ریزشدن دارد که دو شرط زیر را دارا باشد:

* مساحت المان بزرگتر از/2 SF2 باشد (SF تابع اندازه المان می باشد)
* طول هر یک از اضلاع بزرگتر از ضریبی از تابع اندازه المان در وسط المان باشد (در اینجا ضریب برابر 2 است)

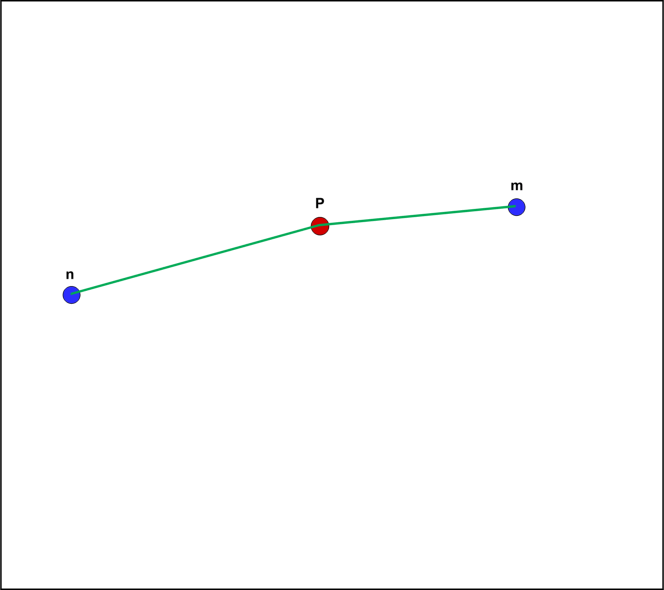
مراحل ریز کردن یک شبکه درشت اولیه و تولید شبکه بی­سازمان دلانی، با استفاده از روش لاسن که در گزارش حاضر پیاده­سازی شده است، بشرح زیر می­باشد:

1. تعیین تابع اندازه المان برای نقاط تشکیل دهنده شبکه درشت
2. تعیین المان که قابلیت ریز شدن داشته و در اولویت است
3. تولید نقطه جدید
4. اتصال رئوس مثلث به نقطه معرفی شده و تشکیل مثلث­های جدید
5. تهیه لیستی از مثلث­های جدید که باید دلانی بودن آنها بررسی گردد
6. بررسی دلانی بودن آخرین مثلث موجود در لیست بالا
7. دلانی کردن مثلث­های غیردلانی موجود در لیست
8. رجوع به مرحله 7 تا زمانی که هیچ مثلثی در این لیست قرار نداشته باشد
9. رجوع به مرحله 3 و انجام هر یک از گام­های تعریف شده تا زمانی که هیچ مثلثی قابلیت دریافت نقطه جدید را نداشته باشد.

در ادامه هر کدام از مراحل فوق بطور مفصل توضیح داده می شود.

### تعیین تابع اندازه المان برای نقاط تشکیل دهنده شبکه درشت

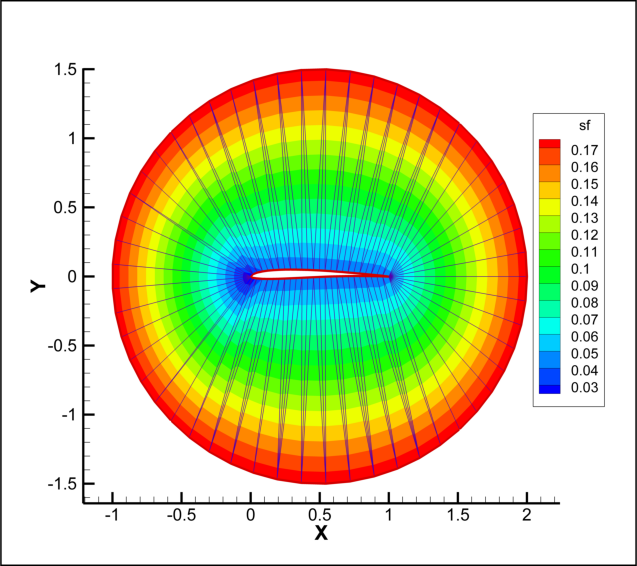
در واقع تراکم مثلث­ها توسط تابع توزیع اندازه المان کنترل می­شود. اگر فاصله نقاط فقط با قرار دادن چشمه در میدان تعریف شود، قدرت و مکان چشمه با سعی و خطا انتخاب خواهد شد که از مشخصه اتوماتیک بودن روش تولید شبکه کاسته می­شود. بنابراین بهتر است که توزیع المان با استفاده از توزیع نقاط مرزی تعیین شود. تابع توزیع اندازه المان با استفاده از توزیع نقاط مرزی و با توجه به فاصله بین یک نقطه و همسایه­هایش مطابق شکل زیر تعیین می­گردد.



1. تابع توزیع اندازه المان تعریف شده برای نقاط مرزی

|  |  |
| --- | --- |
|  |  |

یک نمونه از تابع توزیع اندازه المان که با استفاده از فاصله نقاط مرزی تعیین شده است در شکل زیر نمایش داده شده است. با توجه به شکل مشاهده می­گردد که اندازه المان­ها در نزدیک مرز داخلی کوچک و با پیشروی بسمت مرز خارجی افزایش می­یابد.



1. تعیین تابع توزیع المان بر روی شبکه زمینه

### یافتن المانی که نقطه جدید درون آن قرار دارد

جهت یافتن المانی که قابلیت دریافت نقطه جدیدی را دارد (ریز شدن) دو معیار زیر بکار برده می شود:

• مساحت المان بزرگتر از SF2/2باشد (SF تابع اندازه المان می باشد)

• طول هر یک از اضلاع بزرگتر از ضریبی از تابع اندازه المان در وسط المان باشد (در اینجا ضریب برابر 2 است).

### تولید نقطه جدید

نقطه جدید با استفاده از یک میانگین گیری از مختصات نقاط المانی که باید ریز شود بدست می آید.

### اضافه کردن نقطه معرفی شده به شبکه و تشکیل المان­های جدید

اضافه کردن نقطه جدید به شبکه بطور ساده با متصل کردن نقطه جدید به رئوس المانی که نقطه جدید درون آن قرار دارد، همانند ‏شکل (19) انجام می گیرد.

در اینجا لازم است توجه شود که ترتیب ذخیره نقاط یک مثلث باید بصورت پادساعتگرد باشد. همچنین باید نحوه ذخیره همسایه های بگونه ای باشد که شماره هر همسایه متناظر با راس غیر مشترک باشد. برای روشن شدن این موضوع به ‏شکل (1) توجه کنید.

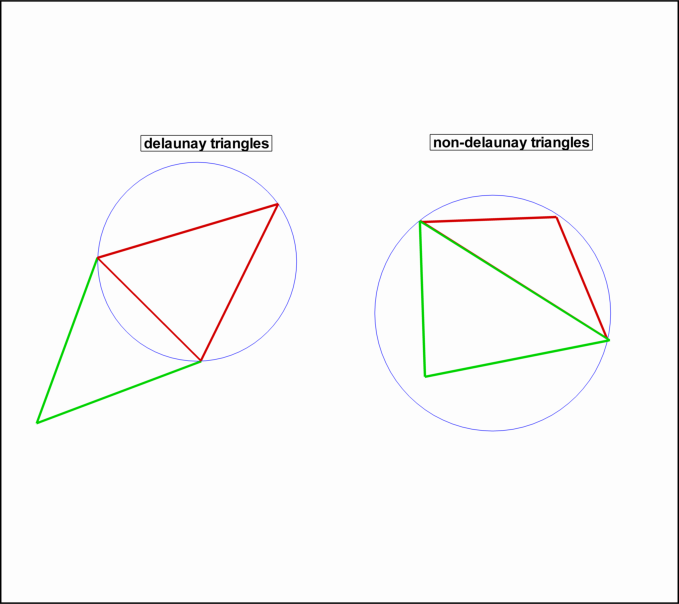
### تهیه لیستی از مثلث­های جدید که باید دلانی بودن آنها بررسی گردد

در ابتدا سه المان تشکیل شده و همسایه آنها در یک لیست ذخیره می گردد تا دلانی بودن آنها بررسی شود. اگر در طی پروسه دلانی کردن این المان ها، المان های جدید تشکیل شود، آنها نیز باید به این لیست اضافه شوند.

### بررسی دلانی بودن آخرین مثلث موجود در لیست بالا

پس از تشکیل المان­های جدید لازم است بررسی شود که آیا المان های جدید دلانی می­باشند یا خیر. بنابراین لیستی از المان­های جدید تشکیل شده و همسایه آنها تهیه می­گردد و دلانی بودن آنها مورد بررسی قرار می­گیرد.

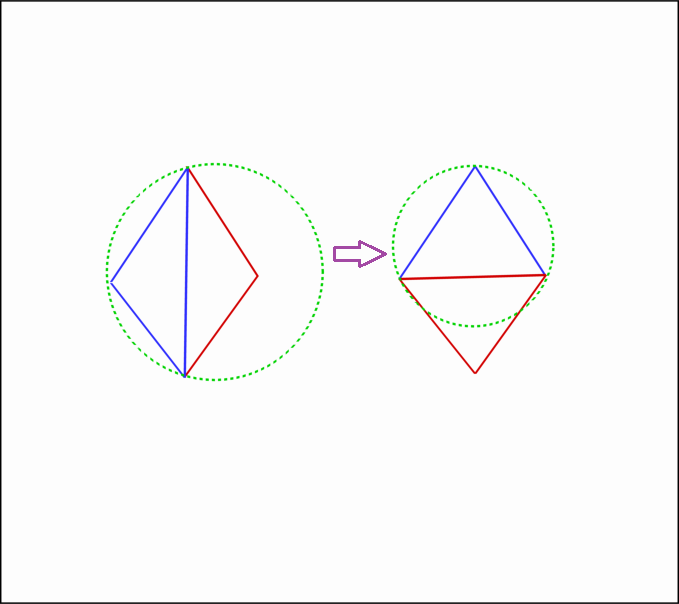
یکی از مهمترین بخش­های این روش تولید شبکه، تشخیص دلانی یا غیر دلانی بودن دو المان همسایه می­باشد. برای این منظور دو المان زیر را در نظر بگیرید. در صورتیکه نقطه P4 درون دایره محیطی مثلث تشکیل شده از نقاط P1/P2/P3 قرار داشته باشد، این دو المان دلانی نیستند. در این حالت با جابجایی ضلع مشترک، این دو المان دلانی می­شوند. همچنین در صورتیکه نقطه P4 بر روی دایره محیطی قرار داشته باشد، این دو المان دلانی می­باشند ولی یکتا نیستند.



1. تشخیص دلانی یا غیر دلانی بودن دو المان در دوبعد

### دلانی کردن مثلث­های غیردلانی موجود در لیست

در فضای دو بعدی با جابجایی ضلع مشترک دو المان غیردلانی همسایه، این دو المان دلانی می‌شوند. برای مثال شکل زیر نحوه انجام این‌کار را نشان می‌دهد.



1. دو مثلث غیردلانی قبل از جابجایی ضلع مشترک

## تبدیل اطلاعات شبکه از حالت سلول محور به ضلع محور

استفاده از ساختار داده‌ای ضلع محور بدلیل کارآمدی آن و همچنین نیاز به حافظه کمتر در ذخیره سازی اطلاعات شبکه بطور گسترده ای مورد توجه قرار گرفته است. از آنجا که در فرایند تولید شبکه از ساختار داده‌ای سلول محور استفاده می‌شود، بنابراین لازم است در انتهای مراحل تولید شبکه ساختار داده‌ای تغییر یابد. همچنین جهت تعیین ضریب فشار و اصطکاک در حلگرهای جریان لازم است از اطلاعات مرزها استفاده شود. علاوه بر این با مرتب کردن اضلاع مرزی می توان در زمان محاسباتی حلگرهای جریان صرفه جویی قابل ملاحظه ای انجام داد. به این دلیل مرتب کردن اضلاع یک شبکه بگونه‌ای که مشخص شود هر کدام از این اضلاع مرزیست یا غیرمرزی و اینکه هر کدام از اضلاع مرزی مربوط به کدام منحنی مرزی می باشد لازم و ضروری می‌باشد.

# پیاده‌سازی و زیربرنامه‌های مورد استفاده

در این فصل، بخش‌های کد و نحوه پیاده‌سازی الگوریتم مرحله به مرحله توضیح داده می‌شود. برای شناخت بهتر زیربرنامه‌های استفاده شده در این برنامه، بهتر است به مستندات آن‌ها مراجعه شود.

1. گرفتن اطلاعات از کاربر

ابتدا با فراخوانی زیربرنامه Read\_2DMeshC فایل ورودی به نام MeshIn.cgid خوانده شده و تمام اطلاعات موجود در این فایل از کاربر گرفته می­شود.

1. چاپ اطلاعات مرزهای شبکه

در این مرحله اطلاعات مربوط به مرزهای شبکه توسط زیربرنامه WriteBoundCrv\_cgid\_plt در یک فایل Plt چاپ می‌شود.

1. تشکیل مثلث بزرگ اولیه

با فراخوانی زیربرنامه Super\_Tri مثلث بزرگ اولیه تشکیل می شود. لازم است توجه شود که در این زیربرنامه مختصات سه نقطه این مثلث محاسبه و به آرایه های مربوطه اضافه می شود. بنابراین سه واحد به تعداد نقاط موجود در میدان اضافه می شود.

1. معرفی نقاط در یک حلقه تکرار

در یک حلقه تکرار تمام نقاط معرفی شده تا به مثلث­بندی موجود اضافه گردد.

1. معرفی نقطه جدید

نقطه جدید در پارامترهای محلی ذخیره شده تا در ادامه برنامه به شبکه موجود اضافه گردد.

1. یافتن مثلثی که نقطه جدید درون آن قرار دارد

با فراخوانی زیربرنامه Tri\_Contain\_Point مثلثی که نقطه معرفی شده درون آن قرار دارد پیدا شده و شماره آن در پارامتر ME ذخیره می­گردد.

1. اضافه کردن نقطه معرفی شده به شبکه

با فراخوانی زیربرنامه Cons\_New\_Tri\_Lawson با متصل کردن رئوس مثلثی که نقطه معرفی شده درون آن قرار گرفته به نقطه معرفی شده، سه مثلث جدید تولید و یک مثلث حذف می­گردد. همچنین لیستی از مثلث­ها که باید دلانی بودن آنها بررسی گردد در این زیربرنامه تولید می­شود.

1. بررسی دلانی بودن مثلث­های جدید

در این بخش در یک حلقه تمام مثلث­هایی که در لیست قرار دارند (مثلث­های جدید) از نظر دلانی بودن بررسی شده و در صورت غیر دلانی بودن، دلانی می­شوند. این حلقه تا جایی ادامه می­یابد که هیچ مثلثی درون لیست قرار نداشته باشد.

1. ذخیره شماره مثلث­ها در پارامترهای محلی

در حلقه موجود همیشه آخرین مثلث موجود در لیست بررسی می­شود. همچنین در یک آرایه دو بعدی که لیست مثلث­ها قرار دارند، مثلث اول بعنوان مثلث اصلی (Main Element) و مثلث دوم بعنوان مثلث همسایه (Neighboring Element) ذخیره می­گردد. همچنین لازم است در ابتدای حلقه تکرار از تعداد مثلث­های موجود در لیست مذکور یک واحد کم شود که در اینجا این کار انجام می­شود.

1. صرفنظر از بررسی المان‌هایی که همسایه نیستند.

ممکن است در پروسه دلانی کردن مثلث ها و با تولید مثلث های جدید، برخی از مثلث های موجود در لیست از بین رفته باشد. بنابراین در صورتیکه دو مثلث موجود در لیست همسایه همدیگر نباشند از بررسی دلانی بودن آنها صرفنظر می شود.

1. بررسی دلانی بودن دو مثلث معرفی شده

با فراخوانی زیربرنامه Del\_Check2D دو مثلثی که در بخش قبل معرفی شده­اند از نظر دلانی بودن بررسی می­گردند.

1. دلانی کردن مثلث­های غیر دلانی

در صورتیکه دو مثلث بررسی شده غیردلانی باشند با فراخوانی زیربرنامه Swapping، دلانی می­شوند.

1. بازیابی اضلاع مرزی

با فراخوانی زیربرنامه BE\_Recovery1\_2D تمام اضلاع مرزی بررسی شده و هر کدام از آن‌ها که در شبکه ورودی موجود نباشد، بازیابی و به شبکه و اطلاعات مرزها اضافه می‌شود.

1. حذف نقاط و المان‌های خارج از میدان(مرز)

با فراخوانی زیربرنامه Delet\_Undesiered\_TriPoint ابتدا تمام المان‌هایی که در طول پروسه تولید شبکه دلانی، خارج از مرزهای بیرونی و یا داخل مرزهای درونی تولید شده بودند به انتهای لیست المان ها انتقال داده می شود. برای حذف این المان‌ها کافیست از تعداد المان‌ها به اندازه تعداد المان‌های قابل حذف کسر شود. همچنین از آنجا که سه نقطه برای تشکیل مثلث بزرگ اولیه تولید شده بود که به انتهای لیست نقاط اضافه شده اند، در اینجا باید این سه نقطه نیز حذف شوند. بنابراین در این مرحله سه واحد از پارامتر مربوط به تعداد نقاط کاسته می شود.

1. چاپ شبکه بعد از اتمام مراحل دلانی، بازیابی اضلاع مرزی و حذف نقاط و المان‌های خارج از میدان

در این بخش ابتدا اضلاع مربوط به هر المان توسط زیربرنامه CellToEdge مشخص می‌شود و سپس شبکه تا این مرحله در یک فایل با فرمت Plt ذخیره می‌شود.

1. شروع فرآیند ریز کردن شبکه به کمک اضافه کردن نقطه و تعیین تابع انداز المان

با فراخوانی زیربرنامه Size\_Function تابع اندازه المان هر کدام از نقاط مرزی که در واقع نقاط تشکیل دهنده شبکه درشت می باشد، تعیین می گردد. در صورتیکه شبکه درشت نقاطی غیر از نقاط مرزی داشته باشد، باید مقدار این تابع بگونه ای دیگر تعیین شود. در اینجا می توان آرایه مربوط به تابع اندازه المان را در عددی ضرب نمود تا تراکم نقاط در کل میدان به دلخواه کاربر کم یا زیاد شود.

1. مقداردهی اولیه به پارامتر Grad

در این بخش مقدار پارامترGrade برای تمام مثلث ها برابر صفر در نظر گرفته می شود تا بررسی درشت بودن یا نبودن آن‌ها در ادامه برنامه انجام شود.

1. تولید نقطه جدید در یک حلقه تکرار

در یک حلقه تکرار درون هر مثلث Ungrade یک نقطه در وسط آن بعنوان نقطه جدید در نظر گرفته می شود و به شبکه معرفی می گردد. این حلقه تکرار به تعداد یک عدد بسیار بزرگ است و در صورتیکه هیچ مثلثی برای دریافت نقطه وجود نداشته باشد با استفاده از دستور Exit از این حلقه خارج می شود.

1. تعیین پارامتر Grad برای مثلث های جدید و پیدا کردن مثلث کاندیدا

مقدار پارامتر Grad برای مثلث‌های جدید تولید شده که مقدار این پارامتر برابر صفر می باشد با فراخوانی زیربرنامه Insid\_Tri\_Grade تعیین می‌گردد. در یک حلقه تکرار مثلث کاندیدا برای نقطه گذاری و پارامتر Grad برای مثلث های جدید تولید شده مشخص می گردد.

1. ذخیره نقاط تشکیل دهنده مثلث کاندیدا در پارامترهای محلی

نقاط تشکیل دهنده مثلث کاندیدا در پارمترهای محلی ذخیره می گردد.

1. تعیین نقطه جدید

پس از تعیین مثلث کاندیدا، یک نقطه در مرکز مختصات آن محاسبه شده تا به لیست نقاط موجود اضافه گردد.

1. اضافه کردن اطلاعات نقطه جدید به لیست نقاط

پس از تعیین نقطه جدید یک واحد به تعداد نقاط موجود اضافه می‌شود و سپس مختصات نقطه وسط مثلث در آرایه‌های مربوط به لیست نقاط موجود اضافه می‌گردد. همچنین مقدار تابع اندازه المان در وسط مثلث کاندیدا ذخیره می‌شود.

1. اضافه کردن نقطه به شبکه موجود

از آنجا که نقطه جدید در وسط مثلث کاندیدا قرار دارد با فراخوانی زیربرنامه Cons\_New\_Tri\_Lawson نقطه جدید به شبکه اضافه می‌شود.

1. تعیین پارامتر Grad برای مثلث های جدید

از آنجا که سه مثلث جدید (مثلث کاندیدا و دو مثلث انتهایی در لیست مثلث های موجود) به شبکه اضافه شده است، مقدار پارامتر Grad برای آنها برابر صفر قرار داده می شود تا در مراحل بعدی مقدار این پارامتر برای آن تعیین گردد.

1. بررسی دلانی بودن مثلث­های جدید

در این بخش در یک حلقه تمام مثلث­هایی که در لیست قرار دارند (مثلث­های جدید) از نظر دلانی بودن بررسی شده و در صورت غیر دلانی بودن، دلانی می­شوند. این حلقه تا جایی ادامه می­یابد که هیچ مثلثی درون لیست قرار نداشته باشد.

1. ذخیره شماره مثلث­ها در پارامترهای محلی

در حلقه موجود همیشه آخرین مثلث موجود در لیست بررسی می­شود. همچنین در آرایه دو بعدیی که لیست مثلث­ها قرار دارند، مثلث اول بعنوان مثلث اصلی (Main Element) و مثلث دوم بعنوان مثلث همسایه (Neighbor Element) ذخیره می­گردد. همچنین لازم است در ابتدای حلقه تکرار تعداد مثلث­های موجود در لیست مذکور یک واحد کم شود که در اینجا این کار انجام می­شود.

1. صرفنظر از بررسی المان‌هایی که همسایه نیستند.

ممکن است در پروسه دلانی کردن مثلث ها و با تولید مثلث های جدید، برخی از مثلث های موجود در لیست از بین رفته باشد. بنابراین در صورتیکه دو مثلث موجود در لیست همسایه همدیگر نباشند از بررسی دلانی بودن آنها صرفنظر می شود.

1. بررسی دلانی بودن دو مثلث معرفی شده

با فراخوانی زیربرنامه Del\_Check2D دو مثلثی که در بخش قبل معرفی شده­اند از نظر دلانی بودن بررسی می­گردند.

1. دلانی کردن مثلث­های غیر دلانی

در صورتیکه دو مثلث بررسی شده غیردلانی باشند (مقدار پارامتر Delauny برابر -1 باشد) با فراخوانی زیربرنامه Swapping، دلانی می­شوند. با دلانی کردن دو مثلث موجود در لیست، دو مثلث جدید جایگزین آنها شده است. بنابراین مقدار پارامتر Grad برای آنها برابر صفر قرار داده می شود تا در مراحل بعدی مقدار این پارامتر برای آن تعیین گردد.

1. بروزرسانی مثلث‌های Ungrade

در یک حلقه تکرار بر روی تمام المان‌های درون میدان در صورتیکه مقدار پارامتر Grad برای یک مثلث برابر -1 باشد به تعداد مثلث‌های Ungrade یک واحد اضافه می‌گردد. لازم بذکر است که این مرحله تنها برای مانیتور کردن پروسه ریز کردن شبکه بکار می رود و تاثیری بر این پروسه ندارد. بنابراین در صورت نیاز می توان این حلقه را اجرا نکرد.

1. تبدیل ساختار اطلاعات شبکه از سلول محور به ضلع محور

در این بخش ساختار داده ای سلول محور که شامل شماره نقاط تشکیل دهنده هر مثلث و همسایه های مثلث ها می باشد گرفته شده و ساختار داده ای ضلع محور بعنوان خروجی زیربرنامه CellToEdge به برنامه اصلی معرفی می شود.

ابتدا در زیربرنامه CellToEdge تمام اضلاع مرزی شناسایی شده و به ابتدای لیست اضلاع منتقل می شود. با توجه به اینکه اضلاع استخراج شده بصورت نامرتب می باشد در اینجا لازم است ترتیب این اضلاع مرتب شده و تعیین شود چند منحنی مرزی وجود دارد و بر روی هرکدام چند ضلع وجود دارد که برای این منظور از زیربرنامه DetectSepetRegnOfMesh2D استفاده می شود.

1. تعریف مقدار برخی از پارامترها

در این بخش برخی از پارامترها همانند ابعاد شبکه، نوع سطوح شبکه (در دوبعدی همان ضلع می‌باشد) و ... مشخص می‌شوند.

1. چاپ نتایج

با استفاده از زیربرنامه WriteMesh\_gid و Write2DMesh\_gid\_plt تمام خروجی­های برنامه در فایل­هایی با فرمت Plt و Gid چاپ می‌شوند.

# مراجع

**[1]** C. L. Lawson, “Software for C 1 surface interpolation,” in Mathematical software, Elsevier, 1977, pp. 161–194.

**[2]** H. Ledoux, “Computing the 3d Voronoi diagram robustly: An easy explanation,” presented at the Voronoi Diagrams in Science and Engineering, 2007. ISVD’07. 4th International Symposium on, 2007, pp. 117–129.

**[3]** J. R. Shewchuk, “Delaunay refinement mesh generation,” CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE, 1997.

**[4]** L. P. Chew, “Guaranteed-quality triangular meshes,” Cornell University, 1989.

1. Cell based [↑](#footnote-ref-1)
2. Edge based [↑](#footnote-ref-2)
3. Lawson [↑](#footnote-ref-3)