

****

**عنوان:**

افزایش کیفیت شبکه مثلثی با تغییر اتصالات(دلانی کردن) با ساختار داده‌ای ضلع محور

|  |  |  |
| --- | --- | --- |
| **نویسندگان** | مرتضی نامور |  |
| احسان فرهادخانی |  |
| **تاریخ تنظیم سند** | 21/11/1394 | |
| **شناسه سند** |  | |

**فهرست مطالب**

[فصل 1- راهنمای کاربری 1](#_Toc512385296)

[1-1- فایل ورودی 1](#_Toc512385297)

[1-2- اجرای برنامه 2](#_Toc512385298)

[1-3- فایل‌های خروجی 2](#_Toc512385299)

[1-4- توانایی‌ها و محدودیت‌ها 2](#_Toc512385300)

[فصل 2- اعتبارسنجی و نتایج 4](#_Toc512385301)

[فصل 3- تئوری و الگوریتم 5](#_Toc512385302)

[3-1- افزودن اضلاع موجود به لیست، جهت بررسی دلانی بودن 6](#_Toc512385303)

[3-2- انتخاب آخرین ضلع موجود در لیست جهت بررسی دلانی بودن 6](#_Toc512385304)

[3-3- بررسی دلانی بودن ضلع انتخاب شده 6](#_Toc512385305)

[3-4- در صورت دلانی بودن : مراجعه به مرحله 2 7](#_Toc512385306)

[3-5- در صورت دلانی نبودن : جابجایی ضلع انتخاب شده 7](#_Toc512385307)

[3-6- اضافه کردن جفت المانهای جدید تشکیل شده به لیست موجود 7](#_Toc512385308)

[3-7- مراجعه به مرحله 2 8](#_Toc512385309)

[فصل 4- پیاده‌سازی و زیربرنامه‌های مورد استفاده 9](#_Toc512385310)

**چکیده:**

در این برنامه یک شبکه مثلثی از کاربر گرفته شده و از نظر دلانی بودن بررسی شده و هر کدام از جفت المان­های آن که غیردلانی باشد، دلانی می­گردد. از آنجا که مثلث بندی دلانی حداقل زاویه المان­های یک شبکه را حداکثر می­کند بنابراین دلانی کردن یک شبکه مثلثی غیردلانی باعث افزایش کیفیت آن می‌گردد. به این دلیل در برخی از مقالات این کار را یکی از روش­های افزایش کیفیت شبکه معرفی کرده­اند. از این برنامه می­توان در روش­های متحرک سازی شبکه استفاده نمود، چرا که با حرکت شبکه کیفیت المان­ها کاهش پیدا می­کند. همچنین پس از پروسه درشت کردن یا ریز کردن شبکه می­توان از این برنامه برای دلانی کردن شبکه جدید تولید شده استفاده نمود. نکته بسیار مهم در این تحقیق اینست که استفاده از ساختار داده‌ای ضلع محور جهت تولید شبکه از نظر محاسباتی بسیار پرهزینه می‌باشد. بنابراین جهت تغییر شبکه محاسباتی مورد استفاده در حلگرهای سیالاتی باید شبکه را ابتدا به ساختار داده‌ای سلول محور منتقل نمود سپس اعمال مطرح در تولید شبکه نظیر ریز و درشت کردن یا دلانی کردن را بر روی آن اعمال و همزمان ساختار داده‌ای ضلع محور را نیز اصلاح نمود.

**کلمات کلیدی:** شبکه مثلثی، دلانی، افزایش کیفیت شبکه.

# راهنمای کاربری

در این برنامه اطلاعات یک شبکه مثلثی بر مبنای ساختار داده ضلع محور از کاربر گرفته می­شود و با دلانی کردن اضلاع تشکیل دهنده شبکه، کیفیت شبکه بر طبق خواص دلانی افزایش پیدا می­کند. دقت شود که از این برنامه تنها برای شبکه­بندی­هایی که المان­های آن مثلثی است، می­توان استفاده کرد. اطلاعات مورد نیاز برای اجرای این برنامه و همچنین خروجی آن را می­توان به صورت زیر دسته بندی کرد.

## فایل ورودی

در این برنامه از یک قالب مشخص برای فایل ورودی که سعی شده در تمامی کدهای حوزه تولید شبکه محاسباتی حفظ شود، استفاده شده است. اطلاعات شبکه ورودی باید در قالب یک فایل و با نام Mesh.gid در پوشه حاوی برنامه قرار داده شود.

اطلاعات زیر باید به ترتیب در فایل ورودی وجود داشته باشد، در غیر اینصورت برنامه خطا خواهد داشت.

1. ابعاد شبکه، دو یا سه بعدی
2. تعداد نقاط تشکیل دهنده شبکه
3. تعداد مثلث­های تشکیل دهنده شبکه
4. تعداد اضلاع تشکیل دهنده شبکه
5. تعداد نواحی شبکه
6. تعداد اضلاع هر ناحیه و نوع آن
7. مشخصات اضلاع شامل المان مجاور سمت چپ، المان مجاور سمت راست، نقطه اول و نقطه دوم
8. مختصات نقاط تشکیل دهنده شبکه

## اجرای برنامه

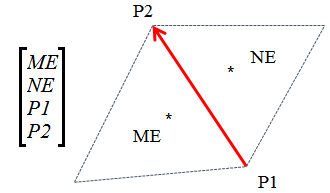
این برنامه تحت Intel fortran و در IDE، Visual studio توسعه داده شده است از این استفاده از این کامپایلر و IDE برای اجرای برنامه پیشنهاد می­شود. در برنامه یک پارامتر به نام Dim به عنوان حداکثر ابعاد آرایه­ها تعریف شده و مقداری پیش­فرض برای آن در نظر گرفته شده است. در صورتیکه در زمان اجرا یا کامپایل برنامه با خطای Array boud exceed مواجه شدید این مقدار را به مقداری مناسب با توجه به ابعاد شبکه افزایش دهید.

## فایل‌های خروجی

برای اینکه بتوان به راحتی بین فایل ورودی و فایل خروجی ارتباط برقرار کرد قالب مشخصی برای فایل خروجی در نظر گرفته شده است. برای اطلاعات بیشتر در مورد ساختار فایل خروجی به مستندات مربوط به زیربرنامه Write2DMesh\_gid\_plt مراجعه کنید. فایل خروجی با فرمت استاندارد نرم­افزار Tecplot و با نام DlnyM.plt قابل مشاهده در این نرم­افزار است. اطلاعات خروجی شامل ساختار شبکه و مرزها است.

## توانایی‌ها و محدودیت‌ها

دو نوع دیدگاه جهت ذخیره اطلاعات شبکه وجود دارد که به آنها اصطلاحا سلول محور[[1]](#footnote-1) و ضلع محور[[2]](#footnote-2) گفته می شود. در اولی نقاط تشکیل دهنده سلول و همسایه های آن ذخیره می گردد و در دومی نقاط تشکیل دهنده ضلع و دو همسایه آن ذخیره می گردد. با توجه به اینکه در دیدگاه ضلع محور نیاز به حافظه کمتر می باشد و همچنین حجم محاسبات کمتری لازم دارد و قابلیت خوبی برای استفاده از شبکه های ترکیبی دارا می باشد، در اینجا از این نوع ساختار داده ای استفاده می شود. با توجه به ‏شکل (1) اطلاعات زیر برای هر کدام از اضلاع تشکیل دهنده شبکه محاسباتی ذخیره می گردد:



1. نحوه ذخیره اطلاعات شبکه

: *ME*سلول سمت چپ (*Main Element*)

: NEسلول سمت راست (Neighboring Element)

: *P1* نقطه ابتدایی

: *P2* نقطه انتهایی

جهت ضلع از اولین نقطه تشکیل دهنده بسمت نقطه دوم می باشد که توجه به آن بسیار ضرویست. سلول اصلی، سلول سمت چپ و سلول همسایه سلول سمت راست ضلع می باشد. در واقع می توان گفت که سلول اصلی سلولی می باشد که ضلع مربوط به آن می باشد و بنابراین محاسبات برای آن انجام می شود.

# اعتبارسنجی و نتایج

جهت اعتبارسنجی و اطمینان از صحت عملکرد برنامه کامپیوتری یک آزمایش بر روی شبکه تولید شده در اطراف یک دایره که درون یک مستطیل قرار دارد انجام گرفته است. همانگونه که در ‏شکل (2) نشان داده شده است کیفیت شبکه در قسمت پایین دایره بسیار پایین است چرا که زوایای مثلث ها بسیار کم می باشد. پس از اجرای برنامه بر طبق خواص دلانی این زوایا بیشتر شده و بنابراین می توان گفت که کیفیت شبکه افزایش پیدا کرده است. در اینجا باید دقت کرد که نحوه پراکندگی نقاط بگونه ای است که نمی توان با این روش افزایش کیفیت، شبکه ای با کیفیت بهتر تولید نمود.



1. شبکه اطراف یک دایره(قبل(قرمز) و بعد(سبز) از افزایش کیفیت)

# تئوری و الگوریتم

افزایش کیفیت شبکه یکی از مراحل مهم تولید شبکه است که در اکثر نرم افزار­های تجاری نیز مورد استفاده قرار می­گیرد. روش­های زیادی برای این کار مورد توجه قرار گرفته شده است. یکی از این روش­ها، تغییر اتصالات است که در آن با جابجایی اتصالات شبکه، یک شبکه دلانی حاصل می­شود. در این گزارش به بررسی نحوه­ی افزایش کیفیت شبکه با استفاده از دلانی کردن آن پرداخته نمی­شود و برای مطالعات بیشتر در این مورد می­توان به مقالات موجود در این زمینه مراجعه نمود. در ادامه به نحوه دلانی کردن یک شبکه غیر دلانی پرداخته خواهد شد.

الگوریتم زیر برای دلانی کردن یک شبکه غیر دلانی مورد استفاده قرار گرفته است:

1. افزودن اضلاع شبکه به یک لیست، جهت بررسی دلانی بودن
2. انتخاب آخرین ضلع موجود در لیست جهت بررسی دلانی بودن
3. بررسی دلانی بودن ضلع انتخاب شده
4. در صورت دلانی بودن : مراجعه به مرحله 2
5. در صورت دلانی نبودن : جابجایی ضلع انتخاب شده
6. اضافه کردن جفت المان­های جدید تشکیل شده و همسایه های آنها به لیست موجود
7. مراجعه به مرحله 2

در ادامه به صورت مفصل به هرکدام از مراحل فوق پرداخته خواهد شد.

## افزودن اضلاع موجود به لیست، جهت بررسی دلانی بودن

با توجه به اینکه وضعیت اضلاع جدید اضافه شده از جهت دلانی بودن نامشخص است برای اینکه دلانی بودن آن­ها بررسی شود به ابتدا لیستی از اضلاع تشکیل می شود که باید دلانی یا غیردلانی اضلاع موجود در این لیست بررسی شود.

## انتخاب آخرین ضلع موجود در لیست جهت بررسی دلانی بودن

از آنجا که اثبات می­شود یکی از خواص شبکه دلانی، یکتا بودن آن است ترتیب بررسی کردن اضلاع اهمیتی ندارد ولی با توجه به اینکه با اصلاح شبکه اضلاع جدید ایجاد و به انتهای لیست اضافه می­شوند انتخاب اضلاع از انتهای لیست بهینه­تر است.

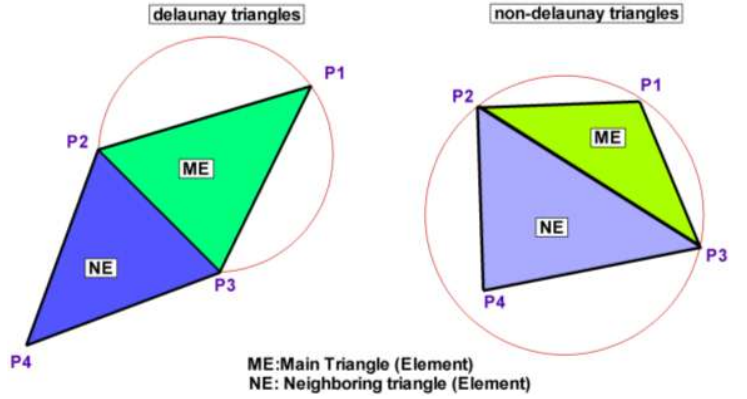
## بررسی دلانی بودن ضلع انتخاب شده

در این مرحله باید مشخص شود که ضلع مورد نظر دلانی است یا خیر. با توجه به ‏شکل (3)، دو المان همسایه در صورتی دلانی هستند که نقطه P4 خارج از دایره محیطی المان اول، یعنی (P1/P2/P3) قرار داشته باشد. برای پیاده­سازی این روش، دترمینان این چهار نقطه محاسبه می­گردد. در صورتیکه المان­ها پادساعتگرد باشند و مقدار دترمینان:

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

دترمینان فوق به صورت رابطه زیر است:

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



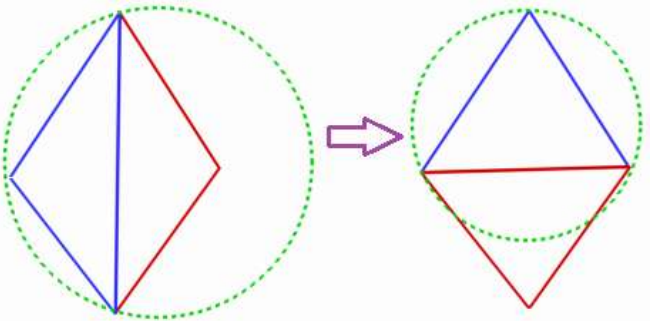
1. مشخص کردن دلانی بودن یا نبودن دو مثلث همسایه

## در صورت دلانی بودن : مراجعه به مرحله 2

در صورتیکه ضلع مورد نظر دلانی باشد ضلع بعدی در لیست بررسی می­شود.

## در صورت دلانی نبودن : جابجایی ضلع انتخاب شده

در صورتیکه ضلع مورد نظر دلانی نباشد با جابجایی، آن ضلع دلانی می­شود. برای مثال ضلع مشترک دو المان سمت چپ ‏شکل (4) با جابجایی (شکل سمت راست) دلانی می شود.



1. نحوه دلانی کردن ضلع غیردلانی

با جابجایی ضلع، المان­های قبلی از بین رفته و مشخصات اضلاع تغییر می­کند.

## اضافه کردن جفت المان­های جدید تشکیل شده به لیست موجود

همانطور که در گام قبل اشاره شد با تغییر ساختار شبکه اضلاع جدید ایجاد شده و اطلاعات برخی اضلاع عوض می­شود از این رو این اضلاع باید برای بررسی مجدد دلانی بودن به لیست اضافه شود.

## مراجعه به مرحله 2

تا زمانی که ضلعی در لیست موجود نباشد باید گام­های 2 تا 7 تکرار شود.

# پیاده‌سازی و زیربرنامه‌های مورد استفاده

در ادامه زیربرنامه­های بکار رفته برای پیاده­سازی الگوریتم تشریح شده در فصل قبل تشریح شده است. در هر زیربرنامه ورودی­ها و خروجی­های آن مشخص شده است. در صورت نیاز به توضیح بیشتر در مورد هر زیربرنامه به گزارش تشریحی آن مراجعه شود.

1. خواندن شبکه­بندی ورودی

با فراخوانی زیربرنامه Read\_2DMesh اطلاعات شبکه از فایل Mesh.gid خوانده می­شود.

1. تشکیل آرایه ارتباط اضلاع و المان­ها

در این گام اضلاع تشکیل دهنده هر المان از شبکه ورودی استخراج شده و با حفظ جهت پادساعتگرد برای استفاده بعدی ذخیره می­شود. هدف از ذخیره سازی این اطلاعات کاهش زمان مورد نیاز برای بهبود شبکه، چک کردن دلانی بودن و اصلاح آن در صورت نیاز است. این کار با فراخوانی تابع EdgeOfCell انجام می­شود. در صورت نیاز به اطلاعات بیشتر به گزارش تشریحی این زیربرنامه مراجعه شود.

1. تهیه لیست اولیه اضلاع برای بررسی دلانی بودن

در این گام لیست اولیه­ای از اضلاع برای بررسی جهت دلانی بودن تهیه می­شود. با توجه به اینکه وضعیت اضلاع از جهت دلانی بودن نامشخص است کلیه اضلاع غیر مرزی به لیست اولیه اضافه می­شوند.

1. چک کردن دلانی بودن اضلاع موجود در لیست و اصلاح آن­ها در صورت نیاز

همانطور که در فصل قبل اشاره شد با توجه به یکتا بودن شبکه دلانی اضلاع موجود در لیست تا زمان خالی شدن آن از جهت دلانی بودن بررسی شده و در صورت نیاز اصلاح می­شوند. بنابراین تمامی اضلاع قرار گرفته شده در آرایه Stack از جهت دلانی بودن بررسی شده و در صورتی که دلانی نباشند با استفاده از زیرتابع SwapEBased2D برای دلانی شدن جابجا می­شوند. این تابع این عمل را تا زمانی که NStack که نشان دهنده تعداد اضلاع داخل آرایه Stack است، صفر شود ادامه می­دهد. با توجه به اینکه برای هر شبکه قطعا یک حالت دلانی وجود دارد نهایتا به این وضعیت خواهیم رسید.

1. انتخاب آخرین عضو آرایه Stack برای چک و دلانی شدن

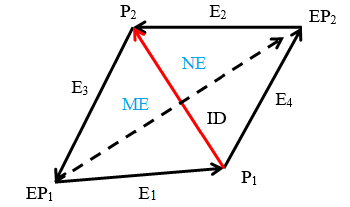
طبق قاعده Stackها آخرین ضلع موجود در آرایه Stack برای چک و دلانی شدن در صورت نیاز انتخاب می­شود. در صورتیکه این ضلع مرزی باشد (المان سمت راست آن صفر باشد) نیازی به چک کردن و اصلاح آن نیست.

1. ذخیره نقاط ابتدا و انتهای ضلع مورد بررسی

نقاط ابتدا و انتهای ضلعمورد بررسی که در واقع همان دو نقطه مشترک دو المان همسایه می باشد در پارامترهای محلی ذخیره می شود.

1. یافتن اطلاعات مورد نیاز برای چک کردن دلانی بودن و اصلاح المان در صورت دلانی نبودن

برای چک کردن دلانی بودن یا نبودن یک ضلع (برای مثال ضلع نشان داده شده با رنگ قرمز در ‏شکل (5)) موقعیت دو نقطه غیر مشترک دو المان همسایه مورد نیاز است. همچنین برای انجام جابجایی ضلع مشترک دو المان مجاور باید شماره اضلاع این دو المان نیز استخراج شود. این کار با فراخوانی زیربرنامه Find\_PointEdge انجام می­شود.



1. اطلاعات مربوط به دو المان همسایه
2. چک کردن دلانی بودن یا نبودن ضلع

با فراخوانی تابع DelCeckEBased2D ضلع مورد نظر از جهت دلانی بودن یا نبودن بررسی می­شود. در صورتیکه المان مورد نظر دلانی باشد DeL=1 خواهد بود و در غیر اینصورت این تابع مقدار DeL را -1 بر می­گرداند.

1. اصلاح ضلع در صورت دلانی نبودن

در صورتیکه در گام قبل تشخیص داده شده باشد که ضلع دلانی نیست با چرخش ضلع به وسیله فراخوانی زیربرنامه SwapEBased2D، وضعیت ضلع اصلاح شده و دلانی می شود. برای توضیح بیشتر به توضیحات زیربرنامه SwapEBased2D مراجعه نمایید.

1. چاپ کردن نتایج

با استفاده از زیربرنامه Write2DMesh\_gid\_plt اطلاعات شبکه­بندی اصلاح شده در فایل خروجی که با فرمت استاندارد Tecplot است، با نام DlnyM.plt چاپ می­شود.

1. Cell base [↑](#footnote-ref-1)
2. Edge base [↑](#footnote-ref-2)