



**زیربرنامه:**

BE\_Recovery1\_2D

|  |  |  |
| --- | --- | --- |
| **توسعه دهنده** | مرتضی نامور |  |
| **تهیه کننده مستند** | مرتضی نامور | |
| **تاییدکنندگان** | مرتضی نامور | |
| **تاریخ تنظیم سند** | 22/02/1394 | |
| **شناسه سند** | **NC5F028F1** | |
| **زبان برنامه‌نویسی** | **Fortran 90/95** | |

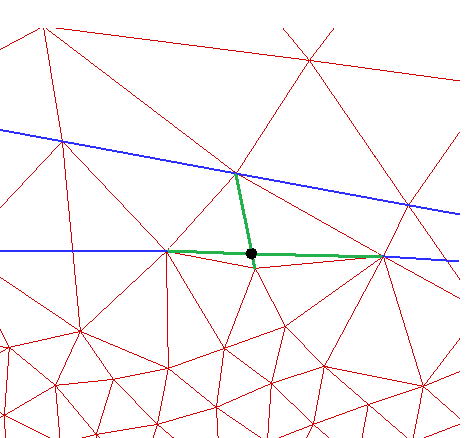
1. وظایف

همانگونه که قبلا اشاره شد روش های دلانی هیچگونه تضمینی برای محافظت از اضلاع مرزی ندارد. بنابراین پس از ایجاد مثلث­بندی دلانی از روی نقاط موجود (مرزی) باید اضلاع مرزی بررسی شده و در صورتیکه این اضلاع تغییر یافته باشند با اضافه کردن نقاط جدید بر روی اضلاع مرزی این اضلاع بازیابی می شوند.

1. توضیحات و تئوری

در ابتدا باید به این نکته توجه کرد که در صورتیکه اضلاع معرفی شده برای انجام یک مثلث بندی دلانی، دلانی باشند می توان تضمین کرد که این اضلاع پس از پایان مراحل مثلث بندی در شبکه تولید شده ایجاد خواهند شد و در صورتیکه اضلاع غیر دلانی باشند حتما این ضلع غیر دلانی معرفی شده به روش تولید شبکه، تشکیل نمی شود. بنابراین باید پس از تولید شبکه با استفاده از نقاط موجود بررسی شود که آیا اضلاع مرزی تشکیل شده است یا خیر. چندین روش برای بازیابی اضلاع مرزی وجود دارد که در این گزارش یکی از ساده ترین آنها پیاده سازی شده است.

در این روش در صورتیکه یک ضلع مرزی تشکیل نشده باشد، ابتدا نقاطه تقاطع آن با اضلاع شبکه محاسبه می شود. سپس این نقطه (که بر روی یکی از اضلاع شبکه قرار دارد) با انجام یک Flip24 به نقاط شبکه اضافه می شود.



1. بازیابی اضلاع مرزی با ایجاد مثلث بندی جدید

این روش بسیار ساده است و یکی از معایب آن اینست که باعث می شود نقاط اضافی[[1]](#footnote-1) بر روی مرزها شبکه ایجاد شود. بنابراین این زیربرنامه برای درک بهتر پروسه بازیابی اضلاع مرزی تدوین شده است و مطالعه آن برای ورود به مباحث مربوط به شبکه سه بعدی (که می توان گفت یکی از حوزه های پژوهشی در زمینه تولید شبکه است) بسیار مفید خواهد بود. همچنین بدلیل محاسبات اندک این زیربرنامه در مقایسه با سایر روش های بازیابی اضلاع مرزی می توان از آن برای حصول اطمینان از تشکیل اضلاع مرزی نیز استفاده کرد.

1. بخش‌های زیربرنامه

در این قسمت، توضیح تمامی بخش‌های زیربرنامه، مطابق شماره‌گذاری انجام شده درمتن برنامه کامپیوتری ارائه شده است.

1. تعیین تعداد اضلاع مرزی

در یک حلقه تکرار بر روی تمام منحنی های مرزی تعداد اضلاع مرزی موجود بر روی هر کدام از منحنی ها به یک شمارنده اضافه می شود تا تعداد اضلاع مرزی تعیین گردد.

1. بررسی اضلاع مرزی مربوط به هرکدام از منحنی های مرزی

در یک حلقه تکرار بر روی تمام منحنی های مرزی و سپس یک حلقه Do while تمام اضلاع مرزی مربوط به هر منحنی بررسی شده و هر کدام از آنها که تشکیل نشده باشد بازیابی می شود. اینکار تا زمانی که شمارنده اضلاع بررسی شده کوچکتر از تعداد اضلاع آن منحنی مرزی باشد ادامه می یابد.

1. ذخیره شماره نقاط تشکیل دهنده ضلع مرزی در پارامترهای محلی

در اینجا شماره نقاط تشکیل دهنده ضلع مورد بحث در پارامتر های محلی ذخیره می­شود. توجه به شماره نقطه اول ضلع و شماره نقطه دوم آن ضروریست.

1. آیا ضلع مرزی تشکیل شده است؟

هر کدام از اضلاع مرزی با تمام مثلث­های درون میدان مقایسه می­شود. بنابراین در اینجا یک حلقه بر روی تمام مثلث­ها وجود دارد. اگر نقطه اول ضلع مرزی مورد بررسی برابر یکی از نقاط تشکیل دهنده مثلث مورد بررسی باشد و نقطه دوم ضلع مرزی برابر نقطه دیگر آن مثلث باشد، ضلع مرزی مورد بررسی تشکیل شده است و با استفاده از دستور Goto به انتهای حلقه Do while می رویم و ضلع مرزی بعدی بررسی می شود.

1. ذخیره مختصات نقاط ضلع مرزی

مختصات نقاط ضلع مرزی در پارامترهای محلی ذخیره می­گردد.

1. پیدا کردن مثلثی که یک نقطه آن با اولین نقطه ضلع مرزی یکسان است

در یک حلقه تکرار تمام مثلث های موجود بررسی می شود و در صورتیکه یک مثلث دارای یک نقطه مشترک با اولین نقطه ضلع مرزی مورد بررسی باشد، باید بررسی شود که آیا ضلع مرزی با ضلع روبری نقطه مشترک تقاطع دارد یا خیر.

1. تعیین مثلث همسایه

با توجه به اینکه نحوه ذخیره همسایه های هر کدام از مثلث ها از یک قاعده خاص پیروی می کند و با توجه به ذخیره شماره سلول ذخیره شده در مرحله قبل، شماره همسایه مثلثی که در مرحله قبل پیدا شده است تعیین می شود. همچنین در این مرحله شماره مثلث پیدا شده در مرحله قبل در یک پارامتر محلی ذخیره می گردد.

1. ذخیره شماره نقاط تشکیل دهند ضلع مشترک دو مثلث همسایه

شماره نقاط تشکیل دهنده ضلع مشترک مثلث تعیین شده در مرحله 6 و مثلث تعیین شده در مرحله 7 در پارامترهای محلی ذخیره می گردد.

1. ذخیره مختصات نقاط ضلع مشترک

مختصات نقاط ضلع مشترک که در مرحله قبل شماره آنها ذخیره شده است، در پارامترهای محلی ذخیره می­گردد.

1. آیا ضلع مرزی و ضلع کاندید از مثلث مورد بررسی تقاطع دارد

با فراخوانی زیربرنامه Cross\_Lines بررسی می شود که آیا ضلع مرزی و ضلع مشترک تقاطع دارد یا خیر. در صورت تقاطع داشتن مقدار پارامتر Intersectبرابر 1 می باشد که در اینصورت با استفاده از دستور Goto جستجو خاتمه می یابد.

1. اضافه کردن نقطه تقاطع به لیست نقاط تشکیل دهنده شبکه

با اضافه کردن یک واحد به شمارنده تعداد نقاط تشکیل دهنده شبکه، مختصات این نقاط در آرایه های مربوطه ذخیره می گردد.

1. اضافه کردن نقطه تقاطع به شبکه

با فراخوانی زیربرنامه Flip24 نقطه تقاطع به شبکه اضافه می گردد. این کار با تقسیم دو مثلث پیدا شده در مرحله 6 و 7 به چهار مثلث انجام می شود.

1. اضافه کردن اضلاع جدید به لیست اضلاع

از آنجا که با اضافه کردن یک نقطه بر روی ضلع مرزی، این ضلع در واقع به دو ضلع تبدیل شده است، باید ضلع مرزی مورد بررسی از لیست مربوطه حذف شود و سپس اضلاع جدید در این لیست قرار داده شود. همچنین بدلیل اینکه شماره اضلاع بصورت مرتب شده به برنامه معرفی شده است باید ابتدا اضلاع به اندازه یک واحد جابجا شود و سپس دو ضلع جدید جایگزین گردد.

پس از جابجایی اضلاع به اندازه یک واحد، ابتدا یک واحد به تعداد تمام اضلاع مرزی و سپس یک واحد به تعداد اضلاع موجود بر روی منحنی مرزی مورد بررسی اضافه می شود. پس از این مرحله دو ضلع جدید به لیست اضلاع اضافه می گردد. توجه شود که جهت این اضلاع مطابق با جهت ضلع مرزی مورد بررسی تعیین شده است.

1. Stainer Points [↑](#footnote-ref-1)