



**زیربرنامه:**

RemoveDiscreteLayerIndexes3D

|  |  |  |
| --- | --- | --- |
| **توسعه دهندگان** | مرتضی نامور |  |
| کامیار صفری |  |
| **تهیه کنندگان مستند** | مرتضی نامور، کامیار صفری | |
| **تاییدکنندگان** | مرتضی نامور | |
| **تاریخ تنظیم سند** | 24/10/1396 | |
| **شناسه سند** |  | |
| **زبان برنامه‌نویسی** | **Fortran 90** | |

1. وظایف

حذف شماره لایه‌ی نقاطی که به صورت اشتباه و گسسته از سایر نقاط و لایه‌ها لایه بندی شده اند.

1. توضیحات و تئوری

در لایه‌بندی‌ای که انجام میشود. دو مشکل ممکن است به وجود آمده باشد.

یکی لایه بندی نقاطیست که هیچ نقطه ای با شماره لایه ی برابر با آن در مجاورتش وجود ندارد. این حالت بیشتر در ناحیه‌ی Wake هستند ممکن است رخ دهد. در این بخش، شماره لایه ی این نقاط برابر با صفر قرار میگیرد.

مشکل دیگر، لایه‌هایی هستند که هیچ اتصالی با دیگر لایه های شبکه و لایه های قبلی خود ندارند. این مشکل در نواحی ای از ناحیه ی همسانگرد که چندین المان کشیده وجود دارد ممکن است رخ دهد. این لایه ها نیز معتبر نبوده و باید شماره لایه ی آنها حذف شود.

بررسی این حالتها در شبکه های سه بعدی بر روی هر سطح شبکه به صورت جداگانه انجام میگیرد. سطوح به ترتیب از نقاط و Faceهای مرزی شروع شده و به ترتیب سایر Faceها را شامل میشود.

1. بخش‌های زیربرنامه

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

1. مقداردهی اولیه متغیرهای مورد استفاده

در این بخش در ابتدا وضعیت تمامی نقاط شبکه را برابر با unknown قرار داده و سپس برای جلوگیری از پردازش مجدد نقاط seed قبلی آنها را برابر با oldSelected قرار میدهیم.

1. اجرای حلقه‌ی اصلی زیربرنامه برای پیمایش سطوح شبکه و سپس پیمایش نقاط

در این بخش با استفاده از یک حلقه‌ی اصلی، در هربار تکرار یکی از سطوح شبکه را پیمایش میکنیم. در هربار تکرار، در یک حلقه داخلیتر تمامی نقاط شبکه پردازش میشوند. این حلقه تا زمانی که حداقل یک تغییر در وضعیت نقاط شبکه به وجود آید ادامه پیدا میکند.

در حلقه‌ی داخلیتر تنها نقاط با وضعیت unknown که بر روی سطح مورد پیمایش قرار دارند مورد بررسی قرار میگیرند.

1. بررسی قرار داشتن نقطه بر روی سطح مورد نظر

با انتخاب هر نقطه، بررسی میکنیم که آن نقطه بر روی سطح مورد پیمایش قرار داشته باشد. زیرا الگوریتم به این گونه است که سطح به سطح بررسی نقاط انجام میگیرد. در صورتی که اولین تکرار حلقه‌ی اصلی باشد، فقط مرزی بودن نقطه بررسی میشود. زیرا که سطح اول شامل نقاط مرزی می باشد. در غیر این صورت بررسی میکنیم که در بین نقاط مجاور نقطه‌ی انتخاب شده حداقل یک نقطه با وضعیت selected وجود داشته باشد.

1. نگهداری نقاط متصل به نقطه‌ی انتخاب شده در هر سطح در یک آرایه کمکی

گفتیم که بررسی اینکه یک نقطه به نقاط دیگری با شماره لایه‌ی برابر متصل است یا خیر، تنها بر روی نقاطی انجام میشود که بر روی یک سطح قرار دارند. بنابراین در این بخش نقاط متصل به نقطه‌ی انتخاب شده که هم سطح آن هستند را در یک آرایه ی کمکی قرار میدهیم.

اینکار به این صورت انجام میگیرد که اگر در حال پردازش سطح اول باشیم، Faceهای مرزی متصل به نقطه‌ی انتخاب شده را بررسی میکنیم و نقاط آن را با آرایه‌ی کمکی اضافه میکنیم.

اما در صورتی که در حال پردازش سطحی غیر از سطح اول باشیم، از لیست نقاط متصل به نقطه‌ی انتخاب شده نقاطی را به آرایه‌ی کمکی اضافه میکنیم که به حداقل یک گره با وضعیت selected متصل باشند.

1. بررسی اتصال نقطه‌ی انتخاب شده به گره‌هایی با شماره لایه‌ی برابر در هر سطح

از بین نقاطی که در مرحله قبل به آرایه کمکی اضافه کردیم، در این بخش دنبال نقاطی با شماره لایه‌ی برابر با شماره لایه‌ی نقطه‌ی انتخاب شده میگردیم.

1. بررسی اتصال نقاط متصل در سطح قبل

با انتخاب هر دو نقطه‌ای که وضعیت آنها unknown است و همچنین به یکدیگر متصل هستند و دارای شماره لایه‌ی برابر هستند، در این بخش نقاط مجاور آنها را در سطح قبل بررسی میکنیم. نقاط مجاور آنها در سطح قبل حداکثر باید با یک واسطه به یکدیگر متصل باشند. در غیر اینصورت نمیتوان دو نقطه را "به هم پیوسته با شماره لایه‌ی برابر در یک سطح" در نظر گرفت. این محدودیت به این دلیل بررسی میشود تا در صورتی که یکی از گره‌های یک سطح میانی با دو سطح مرزی مجاور باشد، نتواند از شرط اتصال به نقاط مرزی با شماره لایه‌ی برابر استفاده کند و این محدودیت را داشته باشیم که هر دو نقطه‌ی مجاور نقطه انتخاب شده موجود بر روی سطح selected، بر روی یکی از رویه‌های مرزی باشند.

در ادامه در صورتی که هیچ ارتباطی بین نقطه‌ی انتخاب شده و نقطه‌ای هم سطح با شماره لایه‌ی برابر وجود نداشته باشد، شماره لایه‌ی مربوط به آن را صفر کرده و آن را از لایه‌بندی حذف میکنیم. همچنین متغیر نشان دهنده‌ی بروز تغییر در شبکه را برابر با true قرار میدهیم تا حلقه اصلی حداقل یکبار دیگر بتواند اجرا شود.

1. نگهداری نقاط فعلی جدید در یک آرایه کمکی

برای اینکه بتوانیم اطلاعات مربوط به وضعیت نقاط سطح جدید را در ادامه بروزرسانی کنیم، لیست نقاط مربوط به سطح فعلی را در یک آرایه کمکی ذخیره میکنیم. سطح فعلی که آن را پردازش کردیم، شامل تمام نقاطیست که وضعیت آنها unknown بوده و به حداقل یک نقطه‌ی با وضعیت selected متصل هستند. همچنین در صورتی که اولین تکرار حلقه باشد، سطح فعلی شامل تمامی نقاط مرزی شبکه با وضعیت unknown می باشد.

1. بروزرسانی وضعیت نقاط سطح پردازش شده و سطح جدید

در این بخش، نقاط selected قبلی را به oldSelected تغییر میدهیم تا دیگر آنها را در پردازشها تاثیر ندهیم. همچنین نقاط سطح فعلی که آن را پردازش کردیم و اندیس آنها را در مرحله قبل در یک آرایه کمکی قرار دادیم را به selected تغییر وضعیت میدهیم.

1. یافتن بزرگترین شماره لایه‌ی موجود

برای اینکه بتوانیم لایه‌ها را با یکدیگر مقایسه کنیم و لایه‌های گسسته را تشخیص دهیم باید تمام لایه‌های شبکه را پردازش کنیم. برای اینکار باید شماره بزرگترین لایه‎ی موجود در شبکه را بدانیم. برای اینکار با استفاده از یک حلقه بزرگترین شماره لایه‌ی موجود در شبکه را پیدا میکنیم و در یک متغیر نگهداری میکنیم.

1. پیمایش لایه‌ها و مقایسه هر لایه با لایه بعد از خودش

هدف از این بخش بررسی این موضوع است که آیا یک لایه با لایه‌ی بعد از خودش اتصال دارد یا خیر. در صورتی که یک لایه با لایه‌ی بعد از خودش اتصال نداشته باشد تمامی لایه‌های از آن لایه به بعد به عنوان لایه‌های گسسته شناخته شده و از لایه‌بندی حذف میشوند.

اتصال به این معنیست که حداقل یک نقطه از لایه‌ی اول به حداقل یک نقطه از لایه‌ی دوم اتصال داشته باشد.

برای اینکار در یک حلقه تمامی لایه‌های شبکه را پیمایش کرده و در هربار پیمایش یک لایه و لایه‌ی بعد از خودش را برای بررسی انتخاب میکنیم.

1. بررسی اتصال دو لایه‌ی انتخاب شده به هم

با انتخاب دو لایه، در یک حلقه تک تک نقاط مربوط به لایه‌ی دوم را انتخاب کرده و به ازای هرکدام از آنها نقاط متصل به آن نقاط را پیمایش میکنیم. در صورتی که شماره لایه‌ی مربوط به هرکدام از این نقاط متصل برابر با شماره لایه‌ی اول باشد، نتیجه میگیریم که دو لایه حداقل یک اتصال داشته و به یکدیگر متصلند. متغیر مورد نظر را با true مقداردهی کرده و از حلقه خارج میشویم.

1. شناسایی شماره آخرین لایه‌ی معتبر

در صورتی که اتصالی بین دو لایه‌ی انتخاب شده وجود نداشته باشد، لایه‌ی انتخاب شده‌ی اول را به عنوان آخرین لایه‌ی معتبر موجود در شبکه در نظر گرفته و آن را در یک متغیر ذخیره کرده و از حلقه خارج میشویم.

1. حذف اندیس لایه از لایه‌های گسسته

در صورتی که در مراحل قبل لایه‌ای به عنوان آخرین لایه شناسایی شده باشد، در این بخش با استفاده از یک حلقه تمامی نقاط دارای شماره لایه‌ی بیشتر از آن را از سیستم لایه‌بندی حذف کرده و شماره لایه‌ی آنها برابر با صفر قرار میگیرد.