به نام خدا

گام زمانی دوگانه

توسعه دهنده:	محمد صابر زمان‌پور زهرائی: کارشناسی ارشد، مکانیک – تبدیل انرژی
تهیه کننده مستند:	محمد صابر زمان‌پور زهرائی
تاریخ تنظیم سند:	۹۴ /۰۳ / ۲۵

فهرست مطالب

مقدمه:

شبیه‌سازی جریان سیال در حالت ناپایا یکی از زمانبرترین و به تبع آن پر هزینه ترین شبیه سازی‌ها در حوزه دینامیک سیالات محاسباتی[1] می‌باشد. برای کاهش زمان اجرای این‌گونه برنامه‌ها روش‌های مختلفی وجود دارد. یکی از پرکاربرد ترین روش‌ها، استفاده از گام زمانی دوگانه[2] می‌باشد.

در واقع با استفاده از دو گام زمانی، یک مسئله ناپایا بصورت پایا حل می¬گردد. به این صورت که در هر گام زمانی واقعی، مسئله بصورت پایا و با استفاده از گام¬های زمانی موضعی حل می¬شود و سپس با همگرا شدن نتایج، یک گام زمانی واقعی پیش می¬رود و به همین منوال در هر گام زمانی واقعی تا همگرا شدن نتایج، از گامهای زمانی موضعی همان گونه که در مسائل پایا به کار می¬رود، استفاده می¬شود که در این صورت با یک مسئله ناپایا با کاربرد دو گام زمانی واقعی و موضعی، بصورت پایا رفتار خواهد شد در این گزارش چگونگی اعمال این روش و نتایج حاصل بحث و بررسی شده است.

[1]CFD
[2]Dual Time Stepping

فصل اول: روش گام زمانی دوگانه

برای شبیه سازی جریان در حالت ناپایا ابتدا باید یک گام زمانی واقعی[3] بـرای اجـرای برنامـه در هـر تکـرار محاسبه شود. این گام زمانی باید به گونه ای انتخاب شود که برای تمام سلول‌ها مناسب باشد.

برای یک سلول دو بعدی با مساحت Area گام زمانی به شکل زیر محاسبه می‌گردد.

$$dt(i,j) = CFL \frac{Area}{(\hat{\Lambda}_c^i + \hat{\Lambda}_c^j)}$$

که CFL توسط کاربر تعیین می‌گردد و Area نیز مساحت سلول می‌باشد. در ادامه داریم:

$$\hat{\Lambda}_c^i = (\left|\vec{v}.\vec{n}^i\right| + c)\Delta S^i$$

$$\hat{\Lambda}_c^j = (\left|\vec{v}.\vec{n}^j\right| + c)\Delta S^j$$

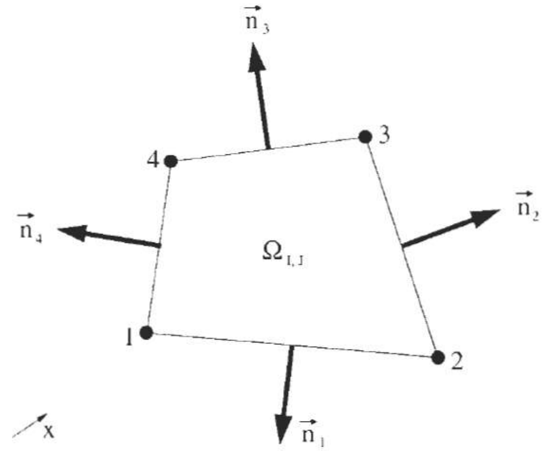
شکل زیر را در نظر بگیرید. آنگاه برای بردارهای نرمال سطح داریم:

$$\vec{n}^i = \frac{1}{2}(\vec{n}_4 - \vec{n}_2)$$

$$\Delta S^i = \frac{1}{2}(\Delta s_4 - \Delta s_2)$$

---

[3] Real Time Stepping



شکل 1 : نمایش بردار نرمال سطح بر روی یک سلول محاسباتی دوبعدی

$$\Delta t = \min(dt(i, j))$$

لازم به ذکر است که گام زمانی واقعی باید به گونه‌ای تعیین شود که خطـوط امـواج در آن بـازه زمـانی همدیگر را قطع نکنند و نیز برای اینکه زمان اجرا طولانی نباشد باید بزرگترین مقدار ممکن را داشته باشـد. این بدان معنی است که بسته به شبکه درنظر گرفته شده برای دامنه حل و اختلاف سـرعت مـوج (مجمـوع سرعت صوت و سرعت جریان) در سلول‌های مختلف، dt برای هر سلول تغییر می‌کند و در نتیجه گام زمانی دستخوش تغییر می‌شود. و هرچه این اختلافات(سرعت موج و مساحت سلول‌ها) بیشـتر باشـد، گـام زمـانی کوچکتر می‌شود و درنتیجه زمان اجرای برنامه طولانی تر می‌گردد. یکی از روش‌های کاهش زمان اجرا، گـام زمانی دوگانه می‌باشد.

معادلات دو بعدی اویلر به صورت زیر می باشد.

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0$$

معادله فوق را می‌توان به صورت زیر نوشت.

$$\frac{Q^{n+1}-Q^{n}}{\Delta t}=-(\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y})$$

که در معادله فوق Δt نشان دهنده زمان واقعی بدست آمده از روابط قبل می‌باشد. این مقـدار ممکـن است برای شبیه‌سازی برخی از جریان‌ها، بسیار کوچک باشد. به همین منظور، زمان مجازی که بسیار بزرگتر از Δt باشد برای شبیه سازی با نام $\Delta\tau$ در نظر گرفته می‌شود. با استفاده از رابطه زیـر مـی‌تـوان گـام زمانی بلندتر برای شبیه سازی استفاده نمود.

$$\frac{Q^{m+1}-Q^{m}}{\Delta\tau}+\frac{Q^{n+1}-Q^{n}}{dt(i,j)}=f(Q)$$

$$f(Q)=-(\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y})$$

در رابطه فوق n شمارنده تکرار اصلی، m شمارنده تکرار داخلی، dt(i,j) زمان بدسـت آمـده بـرای هـر سلول و $\Delta\tau$ زمان انتخاب شده برای پیشروی در هر تکرار اصلی می‌باشد. بـرای اسـتفاده از رابطـه فـوق Q^{n+1} برابر Q^{m+1} قرار داده می‌شود و رابطه زیر بدست می‌آید.

$$Q^{m+1}=\frac{Q^{m}dt(i,j)+Q^{n}\Delta\tau+f(Q)dt(i,j)\Delta\tau}{dt(i,j)+\Delta\tau}$$

در تکرار داخلی اول Q^{m} برابر Q^{n} قرار داده می‌شود. این تکرار تا جایی ادامه مـی‌یابـد کـه لگـاریتم باقیمانده اختلاف دو تکرار کمتر از مقدار خاصی شود. این مقدار می‌تواند بین ۱- تا ۲- باشد.

برای پایداری بیشتر گام زمانی دوگانه، بهتر است از دقت مراتب بالاتر در گام زمـانی اسـتفاده شـود. در روابط فوق از دقت مرتبه ۱ استفاده شده است که می‌توان آن را با دقت مرتبه دو و بالاتر جایگزین نمود.

$$\frac{Q^{n+1}-Q^{n}}{dt(i,j)} \xrightarrow{\text{جایگزین}} \frac{3Q^{n+1}-4Q^{n}+Q^{n-1}}{2dt(i,j)}$$

لازم به ذکر است که برای دقت‌های مرتبه بالا باید از جداسازی "رو به عقب"(BackWard) استفاده شود. برای دقت مرتبه دو، رابطه زیر حاصل می‌شود.

$$\frac{Q^{m+1}-Q^{m}}{\Delta\tau}+\frac{3Q^{m+1}-4Q^{n}+Q^{n+1}}{2dt(i,j)}=f(Q)$$

$$\Rightarrow Q^{m+1}=\frac{2Q^{m}dt(i,j)+4Q^{n}\Delta\tau-Q^{n-1}\Delta\tau+2f(Q)dt(i,j)\Delta\tau}{2dt(i,j)+3\Delta\tau}$$

فصل دوم: شرح برنامه

برای پیاده سازی این روش، از برنامه نوشته شده توسط طهماسبی و قـدک (نـرم افـزار تحلیـل جریـان آرام مبتنی بر معادلات ناویر-استوکس دوبعدی تراکم پذیر ناپایا در جریان داخلی بر روی شبکه سـازمان یافتـه) استفاده شده است.

روش گام زمانی دوگانه، برای برنامه‌های شبیه سازی ناپایا قابل استفاده است و در شبیه سازی حالت پایا این روش کاربردی ندارد. برنامه مورد استفاده به صورت پایا شبیه سازی جریان را انجام می‌دهـد، در ابتـدا ایـن برنامه با تغییراتی که در ادامه بیان شده، به شبیه سازی جریان ناپایا تبدیل می‌شود. سپس روش گام زمـانی دوگانه روی آن اعمال می‌شود.

۲-۱. تبدیل برنامه پایا به ناپایا

تغییرات لازم برای تبدیل برنامه پایا به ناپایا شامل سه بخش است.

۱- قسمت اصلی برنامه(Main)

در این قسمت ۳ متغیر بادقت مضاعف به نام‌های Delt ، TotTime و EndTime تعریف مـی‌شـود. ایـن متغیرها به ترتیب نشان دهنده گام زمانی، زمان در هر لحظه و زمان نهایی شبیه‌سازی می‌باشند.

در قسمت Main Loop ابتدا متغیر TotTime برابر صفر قرار داده می‌شود. این صفر بدان معنی است که شبیه‌سازی از زمان صفر شروع می‌شود. سپس متغیر EndTime مقدار دهی می‌شود. ایـن مقـدار برحسب ثانیه می‌باشد.

شرط حلقه تکرار بـه (N <=Ncycle.AND.TotTime<=EndTime) تغییـر مـی‌یابـد. ایـن شرط بدین معنی است که تا وقتی که زمان کمتر از مقدار نهایی اش است و تکـرار از بیشـینه تکـرار کمتـر است، حلقه ادامه داشته باشد.

در داخل حلقه تکرار بعد از فراخوانی زیر برنامه DeltaTime، گام زمانی محاسبه شده توسط زیر برنامه مذکور به زمان پیش‌روی شبیه سازی اضافه می‌شود.

همچنین بهتر است در قسمت نمایش و چاپ مشخصه‌های اجرا، بجای باقیمانده، مقدار زمان پـیش‌روی چاپ و نمایش داده شود.

۲- زیر برنامه DeltaTime

این زیر برنامه دارای تغییرات جزئی نسبت به حالت پایا می‌باشد. در متغیرهای ورودی این زیر برنامه متغیـر Delt اضافه می‌شود. این متغیر به عنوان یکی از خروجی‌های این زیر برنامه تلقی می‌گردد. در انتهـای ایـن زیر برنامه مقدار کمینه آرایه Dt در متغیر Delt ذخیره می‌شود.

۳- زیر برنامه AUSM

در متغیرهای ورودی این زیر برنامه متغیر Delt اضافه می‌شود. این متغیر به عنوان یکی از ورودی‌های ایـن زیر برنامه تلقی می‌گردد. در انتهای این زیر برنامه کافیست برای محاسبه متغیرDt_Area بجای Dt(i,j) از Delt استفاده می‌شود. در این قسمت از زیر برنامه نیازی به محاسبه باقیمانده نیست.

۲-۲. اعمال گام زمانی دوگانه

تغییرات لازم برای اعمال گام زمانی دوگانه روی برنامه شبیه‌سازی جریان ناپایا، شامل سه بخش می‌باشد.این سه بخش عبارتند از:

۱- قسمت اصلی برنامه(main)

ابتدا شمارنده جدیدی با نام NN تعریف می‌شود که تعداد تکرار حلقه داخلی گام زمانی دوگانه را محاسبه کند. سپس چهار یا هشت آرایه(بسته به نوع گام زمانی دوگانه) به اندازه متغیرهای پایستار معادلات اویلر برای ذخیره سازی گام‌های زمانی قبلی این متغیرها تعریف می‌شود. این آرایه‌ها با نام‌های Q1 ، Q2 ، Q3 و Q4 (و در صورت هشت آرایه ای بودن، چهار آرایه دیگر با نام‌های Q11 ، Q22 ، Q33 و Q44) تعریف می‌شوند.

در حلقه اصلی برنامه مقادیر متغیرهای زمان n (Q1 ، Q2 ، Q3 و Q4) به زمان n-1 (Q11 ، Q22 ، Q33 و Q44) و مقادیر متغیرهای پایستار معادلات اویلر به متغیرهای زمان n (Q1 ، Q2 ، Q3 و Q4) انتقال داده می‌شود. سپس شمارنده حلقه داخلی (NN) و مقدار باقیمانده (Residual) قبل از شروع حلقه داخلی صفر می‌شوند.

در ادامه حلقه داخلی شروع می‌شود. این حلقه دارای شرط اتمام "میزان باقیمانده" است، به این ترتیب که اگر میزان لگاریتم باقیمانده از عدد خاصی کمتر شد حلقه به پایان می‌رسد. در حلقه داخلی فقط زیر برنامه AUSM اجرا می‌شود.

۲- زیر برنامه DeltaTime

در انتهای این زیر برنامه کافیست "Delt=C*Delt"قرار گیرد. C "ضریب زمان" است که به صورت دلخواه توسط کاربر قرار داده می‌شود و گام زمانی را به آن اندازه بزرگتر می‌کند.

۳-زیر برنامه AUSM

در قسمت سوم این زیر برنامه ابتدا مقادیر متغیرهای پایستار برای محاسبه مقدار باقیمانده ذخیره می‌شوند. سپس با استفاده از رابطه زیر (که در فصل قبل به آن اشاره شده بود) مقادیر جدید متغیرهای پایستار معادلات اویلر محاسبه می‌شوند.