



Algoritmo paralelizado en ecuaciones de Navier-Stokes para flujos planos

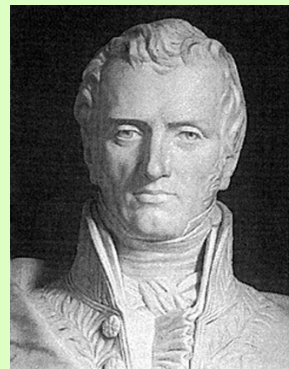
Carlos Torres, Iván Arias, Jefferson Gómez y Sebastián Pabón

Algoritmos computacionales y arquitectura computacional de alto rendimiento

Resumen

Partiendo de las ecuaciones de Navier-Stokes, el desempeño de tareas, procesos y soluciones a problemas aplicados a campos específicos, un lenguaje de programación aprendido y usado durante los cálculos, y el empleo de algoritmos eficientes de caracterizaciones paralelas, se decidió solucionar un ejercicio propuesto relacionado al flujo de un fluido en condiciones especiales a través de una superficie bidimensional, esto, tomando en cuenta parámetros y condiciones de frontera específicas para un caso remoto; sin embargo, su método puede ser replicado para el problema requerido. A continuación se exponen los conocimientos tomados a priori, el desarrollo, y las soluciones.

Ecuaciones de Navier-Stokes



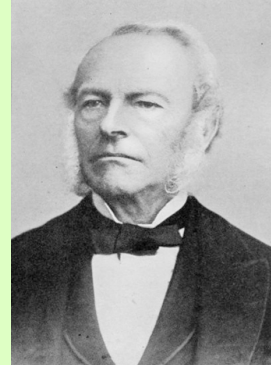
Busto del Claude Louis Marie Henri Navier en la École Nationale des Ponts et Chaussées.

- Aplicación de la segunda ley de Newton

Rapidez de incremento de momentum de una partícula de fluido. = Suma de las fuerzas que están actuando sobre la partícula del fluido.

- Ley de estado que relaciona la presión con la densidad.

- Fluidos compresibles/incompresibles



Sir George Gabriel Stokes.

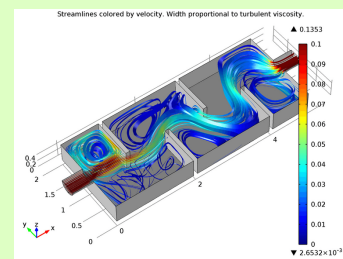
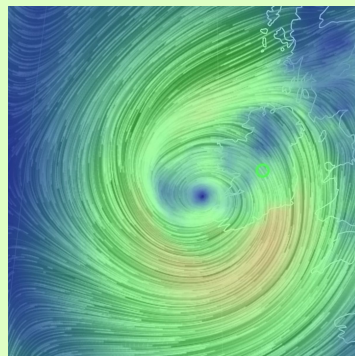
- Modelo de turbulencia
- No linealidad
- Sin solución analítica

Aplicaciones

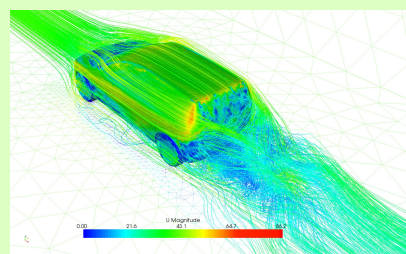


Frentes de aire frío y aire caliente

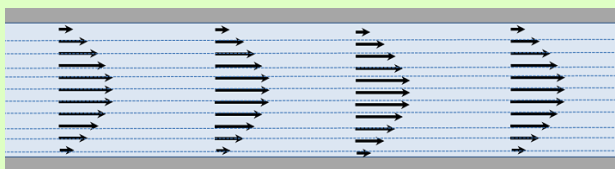
Meteorología



Flujo entre volúmenes/superficies



Aplicación particular



Flujo entre dos placas paralelas (Flujo de Couette)

Ecuaciones generales

Partiendo de la segunda ley de Newton

$$\Sigma \vec{F} = m \vec{a}$$

$$\Sigma \vec{f} = \rho \vec{a} \quad \longrightarrow \quad \vec{a} = \frac{d\vec{v}}{dt} = \frac{\partial \vec{v}}{\partial t} + \frac{\partial \vec{v}}{\partial x} u + \frac{\partial \vec{v}}{\partial y} v + \frac{\partial \vec{v}}{\partial z} w$$
$$\vec{v} = \vec{v}(x, y, z, t) = u(x, y, z, t) \vec{i} + v(x, y, z, t) \vec{j} + w(x, y, z, t) \vec{k}$$

$$\Sigma \vec{f} = \text{Gravedad} + \text{Presión} + \text{Viscosidad}$$

$$\Sigma \vec{f} = \rho \vec{g} - \nabla p + \mu \cdot \nabla^2 \vec{v}$$

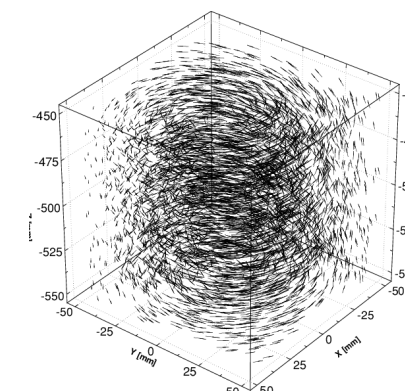
$$\rho \left(\frac{\partial \vec{v}}{\partial t} + \vec{v} \cdot \nabla \vec{v} \right) = \rho \vec{g} - \nabla p + \mu \cdot \nabla^2 \vec{v}$$

Ecuaciones para las componentes dimensionales **x y z**

$$\rho \left(\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} u + \frac{\partial u}{\partial y} v + \frac{\partial u}{\partial z} w \right) = \rho g_x - \frac{\partial p}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right)$$

$$\rho \left(\frac{\partial v}{\partial t} + \frac{\partial v}{\partial x} u + \frac{\partial v}{\partial y} v + \frac{\partial v}{\partial z} w \right) = \rho g_y - \frac{\partial p}{\partial y} + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} \right)$$

$$\rho \left(\frac{\partial w}{\partial t} + \frac{\partial w}{\partial x} u + \frac{\partial w}{\partial y} v + \frac{\partial w}{\partial z} w \right) = \rho g_z - \frac{\partial p}{\partial z} + \mu \left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \right)$$



Campo vectorial tridimensional en "Vórtice de Buggers"

Ecuación de continuidad

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0$$

Ecuaciones de caso particular

$$\begin{aligned} -\frac{\partial P}{\partial r} + \rho g_r &= 0 \\ -\frac{1}{r} \frac{\partial \theta}{\partial r} + \rho g_r &= 0 \\ -\frac{\partial P_z}{\partial z} + \mu \left(\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial v_z}{\partial r} \right) \right) &= 0 \rightarrow -\Delta P + \mu \left(\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial v_z}{\partial r} \right) \right) L = 0 \end{aligned}$$

$$\left(\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial v_z}{\partial r} \right) \right) = \frac{\Delta P}{L\mu} \rightarrow \frac{\partial}{\partial r} \left(r \frac{\partial v_z}{\partial r} \right) = -r \frac{\Delta P}{L\mu}$$

$$\frac{\partial}{\partial r} \left(r \frac{\partial v_z}{\partial r} \right) = -r \frac{\Delta P}{L\mu} \rightarrow r \frac{\partial v_z}{\partial r} = -\frac{r^2}{2} \frac{\Delta P}{L\mu} + c_1$$

$$\frac{\partial v_z}{\partial r} = -\frac{r}{2} \frac{\Delta P}{L\mu} + \frac{c_1}{r} \rightarrow v_z = -\frac{r^2}{4} \frac{\Delta P}{L\mu} + c_1 L n(r) + c_2$$

$$v_z = -\frac{r^2}{4} \frac{\Delta P}{L\mu} + c_2$$

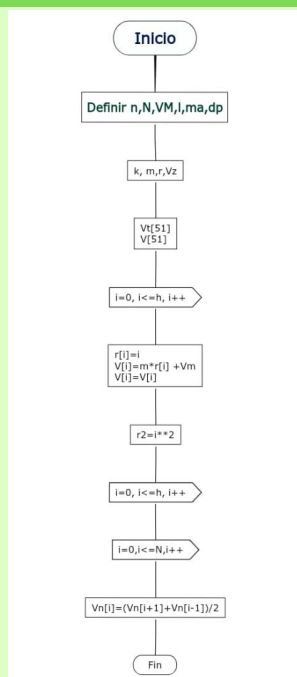
$$v_z(h) = -\frac{r^2}{4} \frac{\Delta P}{L\mu} + c_2 = 0 \rightarrow c_2 = -\frac{h^2}{4} \frac{\Delta P}{L\mu}$$

$$v_z = \frac{\Delta P}{4L\mu} (r^2 - h^2)$$

Referencias

- Temam, R. (2000). Navier-Stokes Equations: Theory and Numerical Analysis. 54-276
- Pereira, F., Stuer, H., Graff, E. and Gharib, M. (2006). Two-frame 3D particle tracking. Measurement Science and Technology. 1686-91
- Girault, V. and Raviart, P. (1979). Finite Element Approximation of the Navier-Stokes Equations. 54-276

Diagrama de flujo



Codificación

```
#include <stdio.h>

int main(int argc, char* argv) {

    float h = 50; // Radio maximo del tubo
    float N = 100; // Cantidad maxima de iteraciones
    float Vm = 5; // Velocidad Maxima
    float l = 100; // Longitud del Tubo
    float mu = 1; // Viscosidad
    float dp = 8; // Cambio de presion
    float k = -dp/(mu*l);
    float m = -Vm/h;
    float r[51];
    float r2[52];

    int nodos = h + 1;

    float Vt[51];
    float V[51];
    double Vn[51];

    int i, q;

    for (i=0; i<=h; i++) {
        r[i] = i;
```

```
        r2[i] = i * i;
        Vt[i] = (k/4) * (r2[i] - h * h);
        V[i] = m * r[i] + Vm;
        Vn[i] = V[i];
    }

    for (q=1; q<=N; q++) {
        for (i=0; i<=h; i++) {
            if (i==0)
                Vn[0] = Vm;
            else
                Vn[i] = 0.5 * ( Vn[i+1] + Vn[i-1]);
        }

        for (i=0; i<=h; i++) {
            printf("%d\t", Vn[i]);
            printf("\b\b\n");
        }

        return 0;
    }
```

«Uno de los valores más preciados de la ciencia es la capacidad de predecir acontecimientos»