GPU Computing (GPGPU) for Computational Fluid Dynamics

Global Scientific Information and Computing Center
Tokyo Institute of Technology

Takayuki Aoki

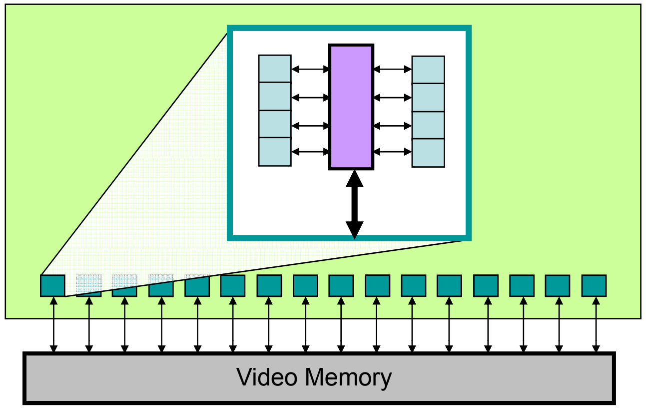
nVIDIA GPU

		GeForce GTX 280 (nVIDIA)
GPU	Peak Performance [GFlops]	622, 933*
	# of SP	240
	SP Clock [MHz]	1296
Video Memory	Transfer Rate[GB/s]	140
	Memory Interface [bit]	512
	Memory Clock[MHz]	2214 (GDDR3)
	Capacity [MB]	1024

•2 instruction issue

Peak Power : 236W

GPU Architecture



■	Global memory	~4GB (video memory: VRAM)
■	Multiprocessor	16(G80, G82), 30(G200)
■	Shared memory	16 Kbyte
□	Streaming Processor	8 SP for 1 Multiprocessor: ~ 240

Types of GPU Usage

on HPC Applications

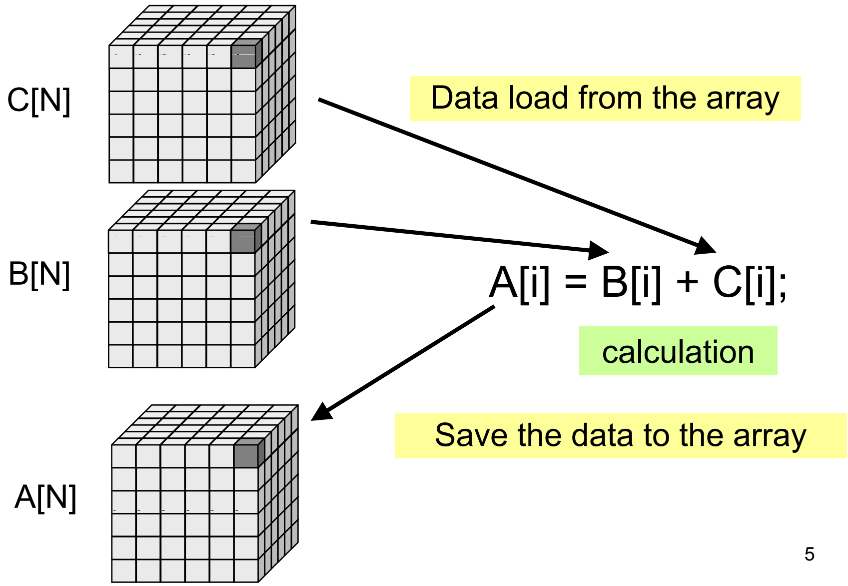
FULL GPU Use **Acceleration ×10～×100**

※ limited types of calculations
※ amount of on-board memory

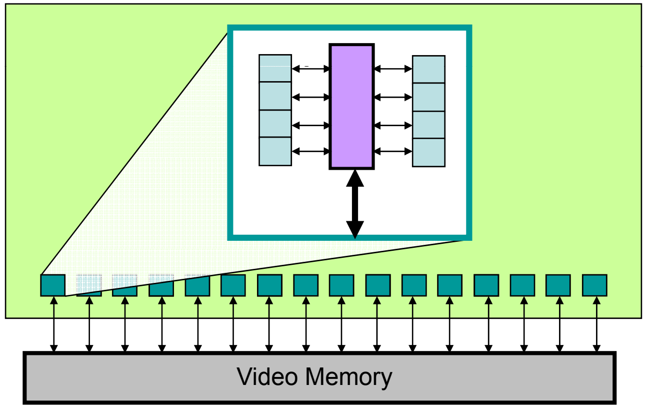
Partial GPU Use **Acceleration 10 % ～ ×3**

※ Hot spot のみ GPU処理
※ Data communication between CPU main memory and GPU VRAM

HPC Computation

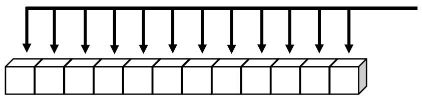


GPU Architecture

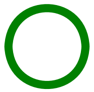


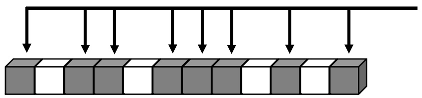
■ (gray)	Global memory	256MB~1GB (video memory: VRAM)
■ (teal)	Multiprocessor	16個(8800GTX(G80, G82))
■ (purple)	Shared memory	16 Kbyte
■ (light blue)	Streaming Processor	8 SP for 1 Multiprocessor: total 128

Types of Memory Access

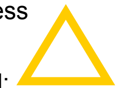


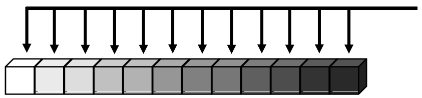
Continuous Access
FDM (Finite Difference)





Random (Indirect) Access
FEM (Finite Element)
A[i] = A[IP[i]] + A[IP[i-1]];





Data Dependency
A[i] = A[i-1] + A[i-2]*C;

Classification of CFD

Compressible fluid analysis

Supersonic flow, Acoustic wave, Explosion, Shock wave, . . .
High-accurate numerical methods:

T. Aoki, Comp. Phys. Comm., Vol.102, No.1-3, 132-146 (1997)
Y. Imai, T. Aoki and K. Takizawa, J. Comp. Phys., Vol. 227, Issue 4, 2263-2285 (2008)
K. Kato, T. Aoki, M. Yoshida, et. al., Int. J. Numerical Methods in Fluids, Vol.51, 1335-1353 (2006)
Y. Imai, T. Aoki, J. Comp. Phys., Vol.217, 453-472 (2006)
Y. Imai, T. Aoki, J. Comp. Phys., Vol.215, 81-97 (2006)

Incompressible fluid analysis

- Most of flow phenomena in our daily life,
- Turbulent flow, Multi-phase flow, Reacting flow, . .
- Semi-implicit Time Integration → Poisson Solver

Incompressible CFD Application

Incompressible Navior-Stokes Equation

$$\nabla \cdot \boldsymbol{u} = 0$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\frac{1}{\rho} \nabla p + \nu \Delta \boldsymbol{u}$$

Poisson equation

$$\Delta p^{n+1} = \frac{\nabla \cdot \boldsymbol{u}^{n+1}}{\Delta t}$$

■ Advection Term :	High-accurate FDM (Suitable for GPU)
■ Diffusion Term :	2nd order Center FDM (easy)
■ Velocity Divergence :	Staggered FDM (easy)
■ Poisson equation :	Red & Black MG (hard)
■ Pressure Gradient :	Staggered FDM (easy)

RIKEN Benchmark Problem

Poisson Equation: (Generalized coordinate)

$$\nabla \cdot (\nabla p) = \rho$$

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} + \frac{\partial^2 p}{\partial z^2} + \alpha \frac{\partial^2 p}{\partial xy} + \beta \frac{\partial^2 p}{\partial xz} + \gamma \frac{\partial^2 p}{\partial yz} = \rho$$

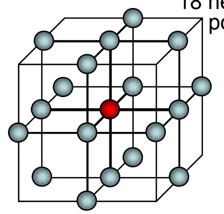
Discretized Form:

$$\frac{p_{i+1,j,k} - 2p_{i,j,k} + p_{i-1,j,k}}{\Delta x^2} + \frac{p_{i,j+1,k} - 2p_{i,j,k} + p_{i,j-1,k}}{\Delta y^2} + \frac{p_{i,j,k+1} - 2p_{i,j,k} + p_{i,j,k-1}}{\Delta y^2}$$

$$+ \alpha \frac{p_{i+1,j+1,k} - p_{i-1,j+1,k} - p_{i+1,j-1,k} + p_{i-1,j-1,k}}{4\Delta x \Delta y}$$

$$+ \beta \frac{p_{i+1,j,k+1} - p_{i-1,j,k+1} - p_{i+1,j-1,k} + p_{i-1,j,k-1}}{4\Delta x \Delta z}$$

$$+ \gamma \frac{p_{i,j+1,k+1} - p_{i,j+1,k-1} - p_{i,j-1,k-1} + p_{i,j-1,k-1}}{4\Delta y \Delta z} = \rho_{i,j,k}$$



Detail of RIKEN Benchmark Problem

Read only 12 arrays : a, b, c, bnd, . . .

Read-write 2 arrays : p, wrk2

```
#define MIMAX            65
#define MJMAX            65
#define MKMAX           129

static float  p[MIMAX][MJMAX][MKMAX];
static float  a[4][MIMAX][MJMAX][MKMAX],
              b[3][MIMAX][MJMAX][MKMAX],
              c[3][MIMAX][MJMAX][MKMAX];
static float  bnd[MIMAX][MJMAX][MKMAX];
static float  wrk1[MIMAX][MJMAX][MKMAX],
              wrk2[MIMAX][MJMAX][MKMAX];
```

```
for(i=1 ; i<imax-1 ; i++)
      for(j=1 ; j<jmax-1 ; j++)
        for(k=1 ; k<kmax-1 ; k++){
          s0 = a[0][i][j][k] * p[i+1][j  ][k  ]
             + a[1][i][j][k] * p[i  ][j+1][k  ]
             + a[2][i][j][k] * p[i  ][j  ][k+1]
             + b[0][i][j][k] * ( p[i+1][j+1][k  ] - p[i+1][j-1][k  ]
                               - p[i-1][j+1][k  ] + p[i-1][j-1][k  ] )
             + b[1][i][j][k] * ( p[i  ][j+1][k+1] - p[i  ][j-1][k+1]
                               - p[i  ][j+1][k-1] + p[i  ][j-1][k-1] )
             + b[2][i][j][k] * ( p[i+1][j  ][k+1] - p[i-1][j  ][k+1]
                               - p[i+1][j  ][k-1] + p[i-1][j  ][k-1] )
             + c[0][i][j][k] * p[i-1][j  ][k  ]
             + c[1][i][j][k] * p[i  ][j-1][k  ]
             + c[2][i][j][k] * p[i  ][j  ][k-1]
             + wrk1[i][j][k];

          ss = ( s0 * a[3][i][j][k] - p[i][j][k] ) * bnd[i][j][k];
          wrk2[i][j][k] = p[i][j][k] + omega * ss;
        }

} /* end n loop */
```

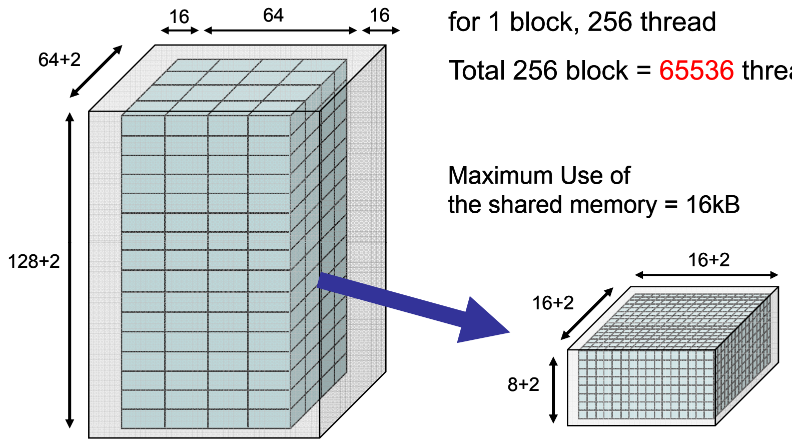

Shared-memory Use

1 block = 16x16x8 : A part of the Computation Domain

for 1 block, 256 thread

Total 256 block = 65536 thread

Maximum Use of
the shared memory = 16kB



Boundary : Non-coal eased data access

Memory Bounded Problem

A[129][65][65] : 2.18 MB × 14 variables

12 : read only
1 : read-write
1 : write

```
#define MIMAX            65
#define MJMAX            65
#define MKMAX            129

static float  p[MIMAX][MJMAX][MKMAX];
static float  a[4][MIMAX][MJMAX][MKMAX],
              b[3][MIMAX][MJMAX][MKMAX],
              c[3][MIMAX][MJMAX][MKMAX];
static float  bnd[MIMAX][MJMAX][MKMAX];
static float  wrk1[MIMAX][MJMAX][MKMAX],
              wrk2[MIMAX][MJMAX][MKMAX];
```

per 1 grid = 34 floating point calculations

per 1 word Data transfer = 34/14 = 2.4

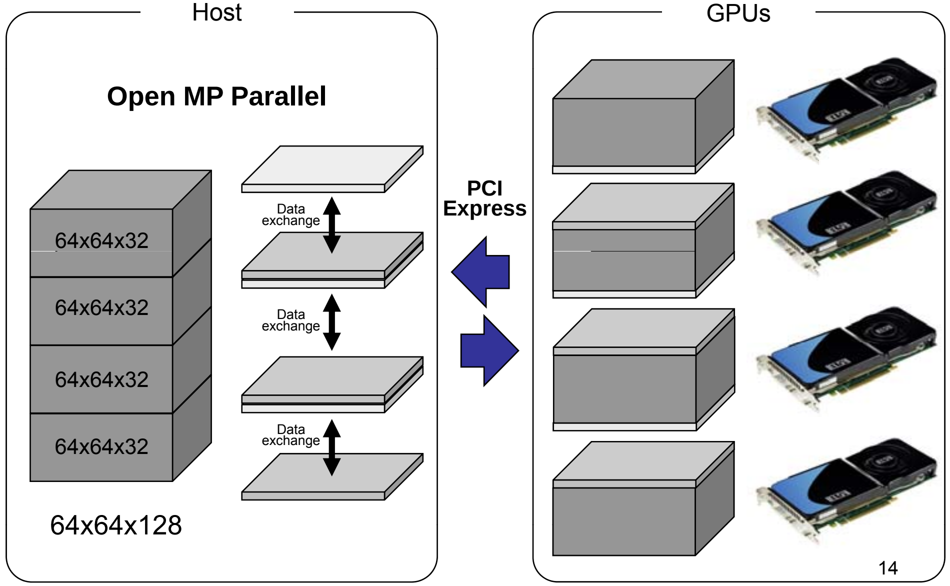
If GPU data transfer rate is
60 GB/sec（15 GWord/sec）,

Even if GPU is very fast

15 × 2.4 = 36.4 GFLOPS

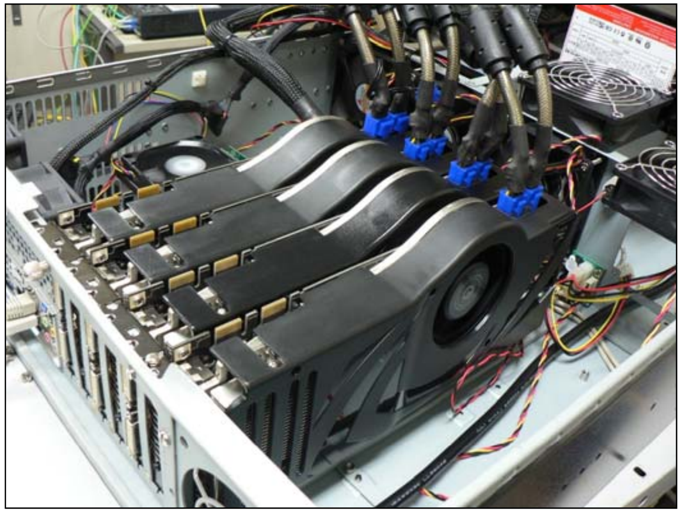
Without shared memory : 34/14 → 34/(14+18) = 1.06
15 × 2.4 = 15.9 GFLOPS

Parallelization using 4 GPU



Machine (Phenom Desktop PC ＋ 4 GPU)

GeForce 8800 Ultra x 4



Specifications of GPU and Motherboard

		GeForce 8800Ultra(G80) (MSI)	GeForce 8800Ultra(G80) (ELSA)
GPU	Peak Performance [GFlops]*	414.2	384
	# of SP	128	128
	SP Clock(CoreCock)[MHz]	1618(660)	1500(612)
Video Memory	Transfer Rate[GB/s]	110.4	103.68
	Memory Bus width[bit]	384	384
	Data Rate[GHz]	2.3(GDDR3)	2.16(GDDR3)
	Capacity [MB]	768	768

* 2 instruction issue

MSI K9A2 Platinum
AMD 790FX + AMD SB600
4 PCI-Express x16-type slots
•2 slot Support up to PCI-Express 2.0 x 16
•2 slot Support up to PCI-Express 2.0 x 8

When using 4 GPU card, it works as PCI-Express x 8 for each GPU

RESULT

0.976 GFLOPS (8.431sec) → 51.91 GFLOPS (0.158sec)

× 53.1

- Before

```
mimax = 65 mjmax = 65 mkmax = 129
imax = 64 jmax = 64 kmax =128
 Start rehearsal measurement process.
 Measure the performance in 3 times.

 MFLOPS: 941.082902 time(s): 0.052496 3.288628e-03

 Now, start the actual measurement process.
 The loop will be excuted in 500 times
 This will take about one minute.
 Wait for a while

 Loop executed for 500 times
 Gosa : 9.673350e-04
 MFLOPS measured : 976.566479        cpu : 8.431426
 Score based on Pentium III 600MHz : 11.909347
```

- After

```
[INFO] Number of host available CPU : 4
[INFO] Number of CUDA devices       : 4

 4-GPU OpenMP execution . . . . .

mimax = 65 mjmax = 65 mkmax = 129
imax = 64 jmax = 64 kmax =128
 Start rehearsal measurement process.
 Measure the performance in 3 times.

 MFLOPS: 34717.560084 time(s): 0.001423 3.295089e-03

 Now, start the actual measurement process.
 The loop will be excuted in 500 times
 This will take about one minute.
 Wait for a while

 Loop executed for 500 times
 Gosa : 9.672065e-04
 MFLOPS measured : 51909.594689      cpu : 0.158619
 Score based on Pentium III 600MHz : 633.043838
```


Parallel Performance

S model [65x65x129]

1 GPU (no data transfer)	**30.6 GFLOPS**	**(0.269sec)**
2 GPU (16kB transfer)	**42.5 GFLOPS**	**(0.193sec)**
4 GPU (32kB transfer)	**51.9 GFLOPS**	**(0.158sec)**

Reference

M model [129x129x257]

1 GPU (no data transfer)	29.4 GFLOPS	(2.328sec)
2 GPU (66kB transfer)	53.7 GFLOPS	(1.275sec)
4 GPU (131kB transfer)	83.6 GFLOPS	(0.819sec)

L model [257x257x512]

1 GPU (no data transfer)	....	
2 GPU (262kB transfer)	....	
4 GPU (524kB transfer)	**93.6 GFLOPS**	(5.974sec)

Poisson Equation solved by

MG(Multi Grid), Red & Black method

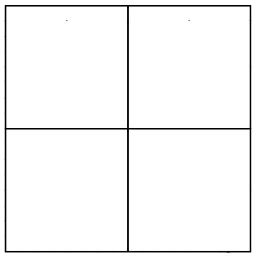
■ **Algorithm Acceleration**

Point Jacobi $\xrightarrow{\times 4 \sim 5}$ SOR $\xrightarrow{\times 100}$ MG-SOR

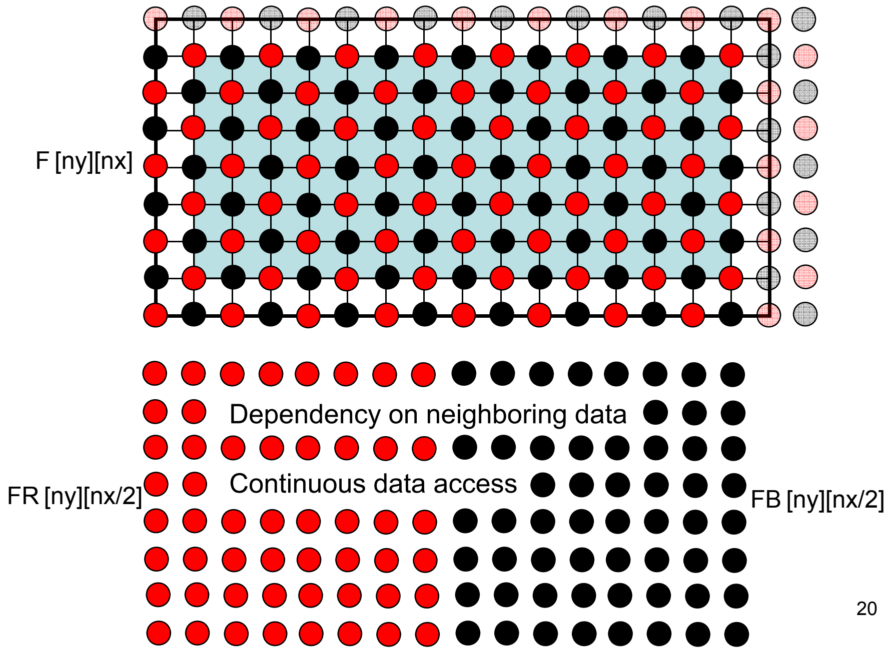
■ **Hardware Acceleration :**

GPU (CUDA) ×50?

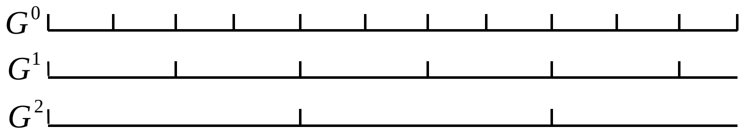
$$\frac{f_{i+1,j}-2f_{i,j}+f_{i-1,j}}{\Delta x^2}+\frac{f_{i,j+1}-2f_{i,j}+f_{i,j-1}}{\Delta y^2}=\rho_{i,j}$$



Red & Black method



Multi-Grid Method



Δx^k Grid spacing of the k-th Grid G^k $\qquad \Delta x^k = 2\Delta x^{k-1}$

Discretized Poisson Equation on G^k

$$L^k F^k = S^k$$

L^k : operator $\quad F^k$: exact solution

S^k: source term

$$f_1^k = SOR_{RED\&BLACK}\left(L^k, S^k, f_0^k, n\right)$$

n-times iteration result, starting from f_0^k

Residual $R^k = S^k - L^k f_1^k$

Correction Equation

Correction $\quad v^k = F^k - f_1^k$

Residual $\quad R^k = S^k - L^k f_1^k$

$$= L^k F^k - L^k f_1^k = L^k\left(F^k - f_1^k\right)$$

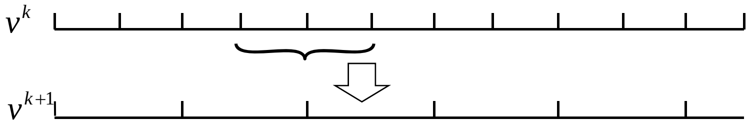
$$\therefore \quad L^k v^k = R^k$$

The k-th grid correction equation has the same form as Poisson equation.

$$L^{k+1} v^{k+1} = R^{k+1} \qquad \left(\Delta x^{k+1} = 2\Delta x^k\right)$$

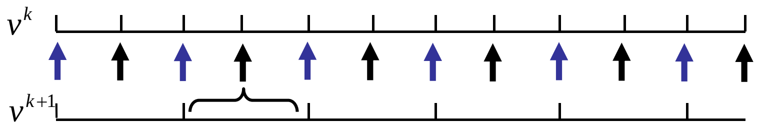
The $k+1$-th grid correction equation

Restriction and Prolongation



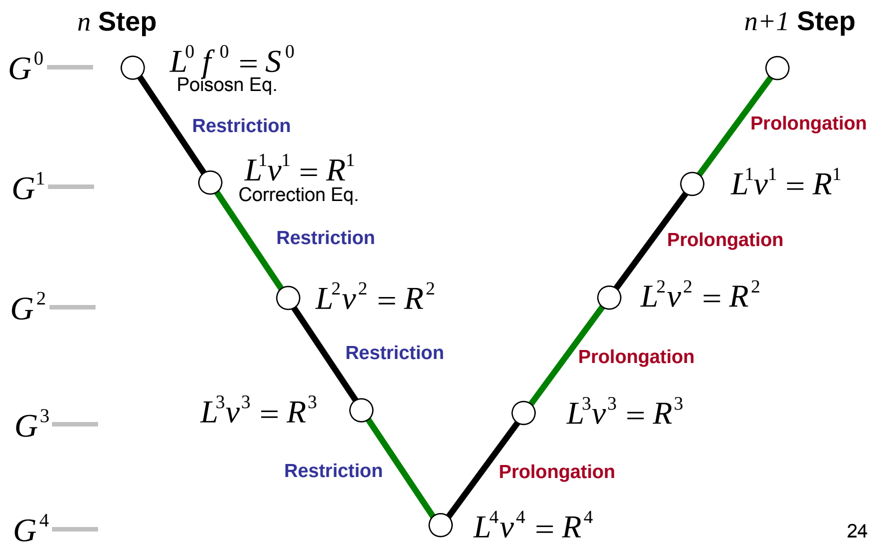
$$v_i^{k+1} = \frac{1}{4}\left(v_{i+1}^k + 2v_i^k + v_{i-1}^k\right)$$

Solving $k+1$-th correction equation $L^{k+1} v^{k+1} = R^{k+1}$



$$v_i^k = \frac{1}{2}\left(v_{2i+1}^{k+1} + v_{2i}^{k+1}\right)$$

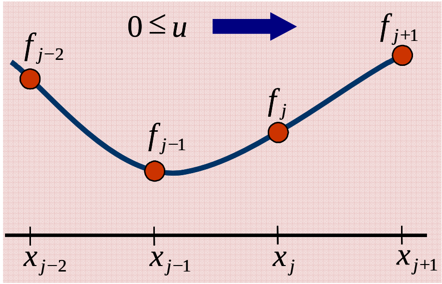
V-Cycle MG



Two-dimensional Advection Equation

Most basic hyperbolic equation:

$$\frac{\partial f}{\partial t}+u\frac{\partial f}{\partial x}+v\frac{\partial f}{\partial y}=0$$



Cubic Semi-Lagrangian Scheme (3rd-order accuracy in time and space)

$$f_j^{n+1}=F_c^n(x_j-u\Delta t)=a(-u\Delta t)^3+b(-u\Delta t)^2+c(-u\Delta t)+f_j^n$$

$$a=\frac{f_{j+1}^n-3f_j^n+3f_{j-1}^n-f_{j-2}^n}{6\Delta x^3}\quad b=\frac{f_{j+1}^n-2f_j^n+f_{j-1}^n}{2\Delta x^2}\quad c=\frac{2f_{j+1}^n+3f_j^n-6f_{j-1}^n+f_{j-2}^n}{6\Delta x}$$

Two-dimensional Advection Equation

Frontgenesis velocity profile:

GeForce 8800 GTS

65 GFLOPS

1024 × 1024

Two-Stream Instability
in Plasma Physics

Vlasov-Poisson Equation:

$$\frac{\partial f}{\partial t}+v\frac{\partial f}{\partial x}-\frac{eE}{m_e}\frac{\partial f}{\partial v}=0 \qquad \frac{\partial^2\phi}{\partial x^2}=\frac{e(n_e-n_i)}{\varepsilon_0}$$

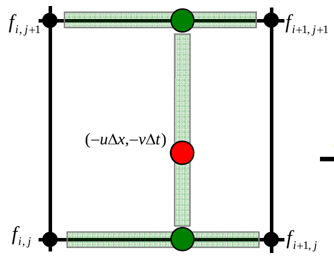
$$\left(E=-\frac{\partial\phi}{\partial x},\quad n_e=\int f dv\right)$$

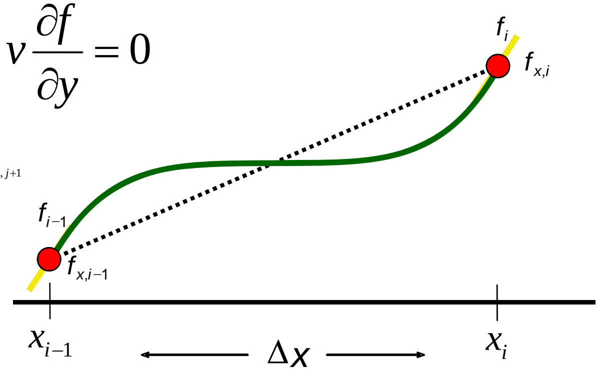
f **: electron distribution function**

n **: electron number density**

CIP Method for 2-dimensional Advection Equation

$$\frac{\partial f}{\partial t}+u\frac{\partial f}{\partial x}+v\frac{\partial f}{\partial y}=0$$

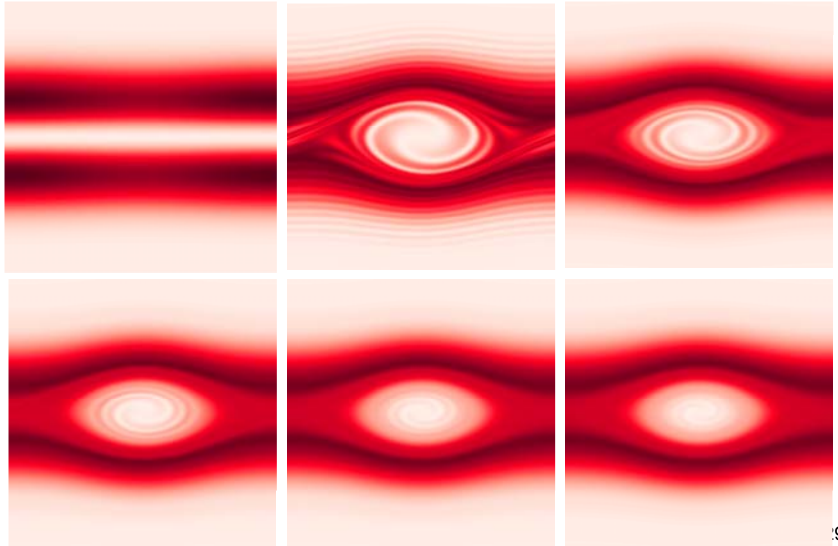




$$f_i^{n+1}=F_{CIP}(-u\Delta x)=a\xi^3+b\xi^2+f_{x,i}\xi+f_i$$

$$a=\frac{1}{\Delta x^2}\left(f_{x,i}+f_{x,i-1}\right)-\frac{2}{\Delta x}\left(f_i-f_{i-1}\right),\quad b=\frac{1}{\Delta x}\left(2f_{x,i}+f_{x,i-1}\right)-\frac{3}{\Delta x^2}\left(f_i-f_{i-1}\right)$$

120 GFLOPS using 8800GTS

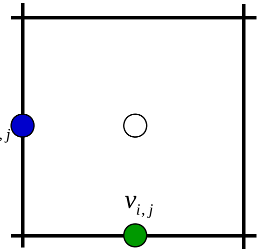


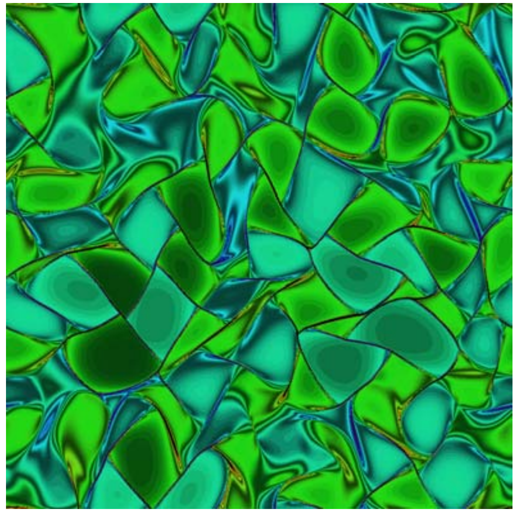
Two-dimensional Burgers Equation

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=\kappa\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)$$

$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}=\kappa\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)$$

GeForce 8800 GTS
40 GFLOPS





1024 × 1024

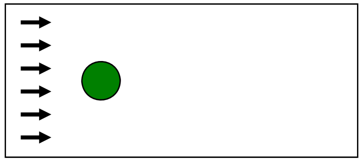
v の定義点における速度 u $\quad u_s=\dfrac{u_{i,j}+u_{i+1,j}+u_{i,j-1}+u_{i+1,j-1}}{4}$

Incompressible Flow around a Body

Incoming Flow: uniform

Karman Vortex street behind the body depending Reynolds number

Re = 2000



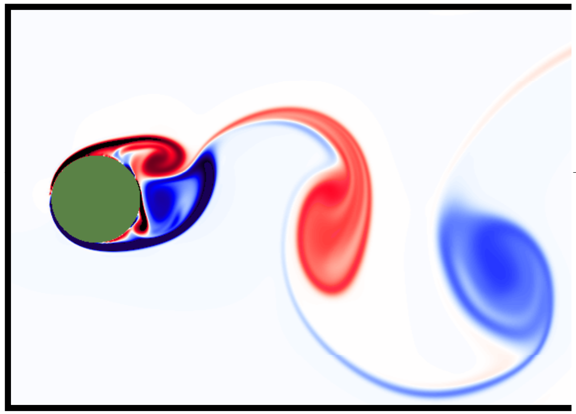
Burgers Equation

Poisson Equation

$$\frac{\partial^2 p}{\partial x^2}+\frac{\partial^2 p}{\partial y^2}=\frac{1}{\Delta t}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right)$$

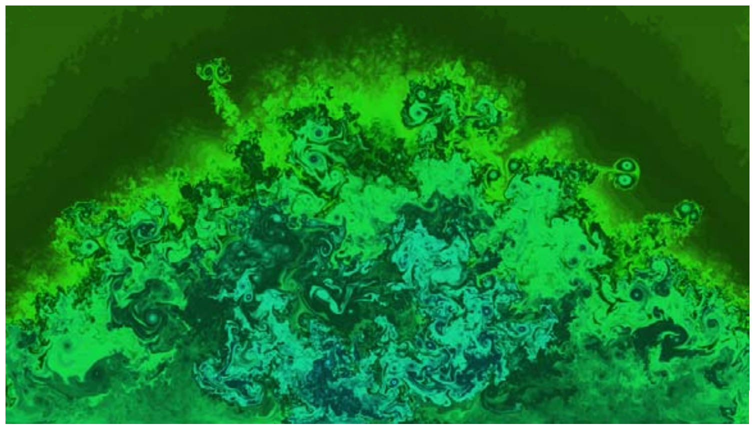
Correction

$$\frac{\partial u}{\partial t}=-\frac{1}{\rho}\frac{\partial p}{\partial x} \qquad \frac{\partial v}{\partial t}=-\frac{1}{\rho}\frac{\partial p}{\partial y}$$



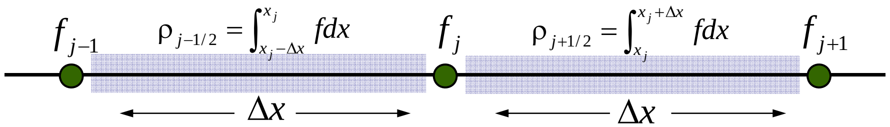
Compressible CFD Application

High-accurate numerical Scheme becomes very important.



Numerical Scheme IDO-CF

Y. Imai, T. Aoki and K. Takizawa, J. Comp. Phys., Vol. 227, Issue 4, 2263-2285 (2008)



$$F(x) = ax^4 + bx^3 + cx^2 + dx + f_j$$

Four matching conditions : $\int_{x_j-\Delta x}^{x_j} F(x)dx = \rho_{j-1/2}$ $\quad F(-\Delta x) = f_{j-1}$ $\quad F(\Delta x) = f_{j+1}$, $\quad \int_{x_j}^{x_j+\Delta x} F(x)dx = \rho_{j+1/2}$

Unknown coefficients : $c = \frac{5}{4}\frac{3\rho_{j+1/2}+3\rho_{j-1/2}-6f_j\Delta x}{\Delta x^3} - \frac{3}{4}\frac{f_{j+1}-2f_j+f_{j-1}}{\Delta x^2}$ $\quad d = 2\frac{\rho_{j+1/2}-\rho_{j-1/2}}{\Delta x^2} - \frac{f_{j+1}-f_{j-1}}{2\Delta x}$

$$\frac{\partial}{\partial x}F(0) = 2\frac{\rho_{j+1/2}-\rho_{j-1/2}}{\Delta x^2} - \frac{f_{j+1}-f_{j-1}}{2\Delta x}$$

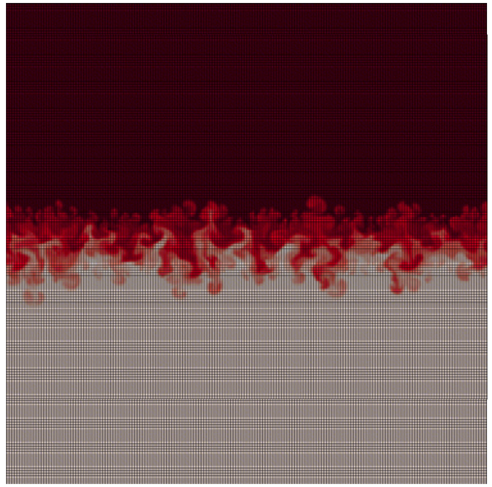
$$\frac{\partial^2}{\partial x^2}F(0) = \frac{5}{2}\left(\frac{3\rho_{j+1/2}+3\rho_{j-1/2}-6f_j\Delta x}{\Delta x^3}\right) - \frac{3}{2}\left(\frac{f_{j+1}-2f_j+f_{j-1}}{\Delta x^2}\right)$$

Rayleigh-Taylor Instability

Heavy fluid lays on light fluid and unstable.

× 90

512 x 512



Euler equation:

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

$$\mathbf{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix} \quad \mathbf{E} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ eu + pu \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ ev + pv \end{bmatrix}$$

42 GFLOPS using GTX280

Phase Separation

Phase transition dynamics is described by the Phase Field Model.

Cahn-Hilliard equation:

$$\frac{\partial \psi}{\partial t} = L\nabla^2\left(\frac{\partial H}{\partial \psi} - C\nabla^2\psi\right)$$

H : free energy

$$\frac{\partial H}{\partial \psi} = \tau\psi - u\psi^3$$

Discretization:

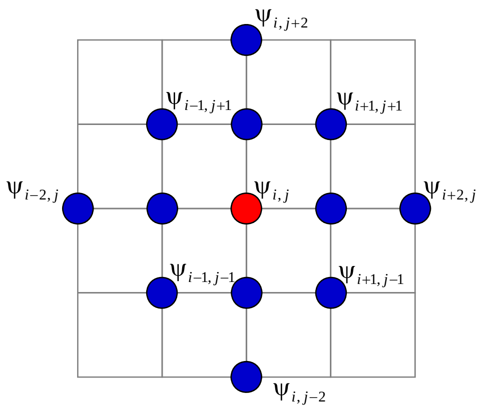
$$\frac{\partial^4 \psi}{\partial x^4} = \frac{\psi_{i+2,j} - 4\psi_{i+1,j} + 6\psi_{i,j} - 4\psi_{i-1,j} + \psi_{i-2,j}}{\Delta x^4}$$

$$\frac{\partial^4 \psi}{\partial x^2 \partial y^2} = \big(\psi_{i+1,j+1} - 2\psi_{i,j+1} + \psi_{i-1,j+1} - 2\psi_{i+1,j} + 4\psi_{i,j} - 2\psi_{i-1,j} + \psi_{i+1,j-1} - 2\psi_{i,j-1} + \psi_{i-1,j-1} \big)$$

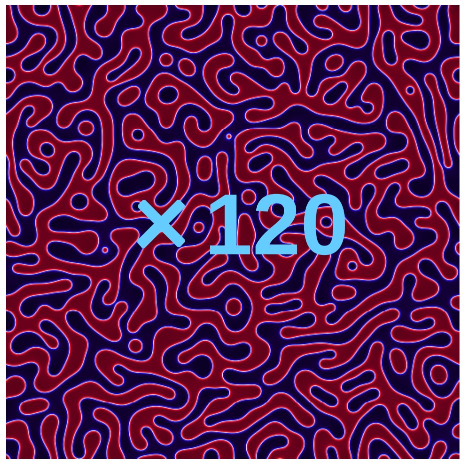
2-D Computation of Phase Separation

Mixture of Oil and Water:

114 GFLOPS using GTX280



512 x 512



3-D Computation of

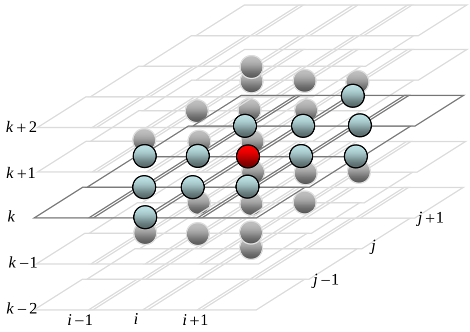
Phase Separation

Mixture of Oil and Water: **158 GFLOPS** using GTX280

Used register number = 46

※ nvcc option –maxrregcount 32

for G80, 92



256 x 256 x 256