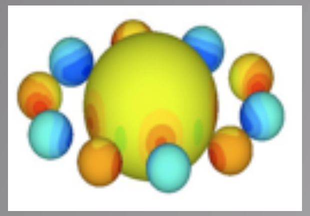
12 Steps to a Fast Multipole Method on GPUs

Rio Yokota
Boston University

Application of N-body methods

Poisson

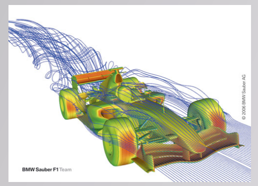
$$\nabla^2 u = -f$$



Astrophysics

Electrostatics

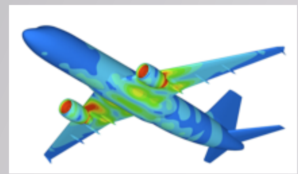
Fluid Mechanics



Helmholtz

$$\nabla^2 u + k^2 u = -f$$

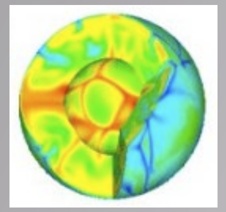
Acoustics



Electromagnetics

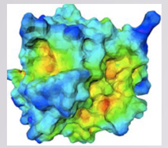
Poisson-Boltzmann

$$\nabla \cdot (\epsilon \nabla u) + k^2 u = -f$$



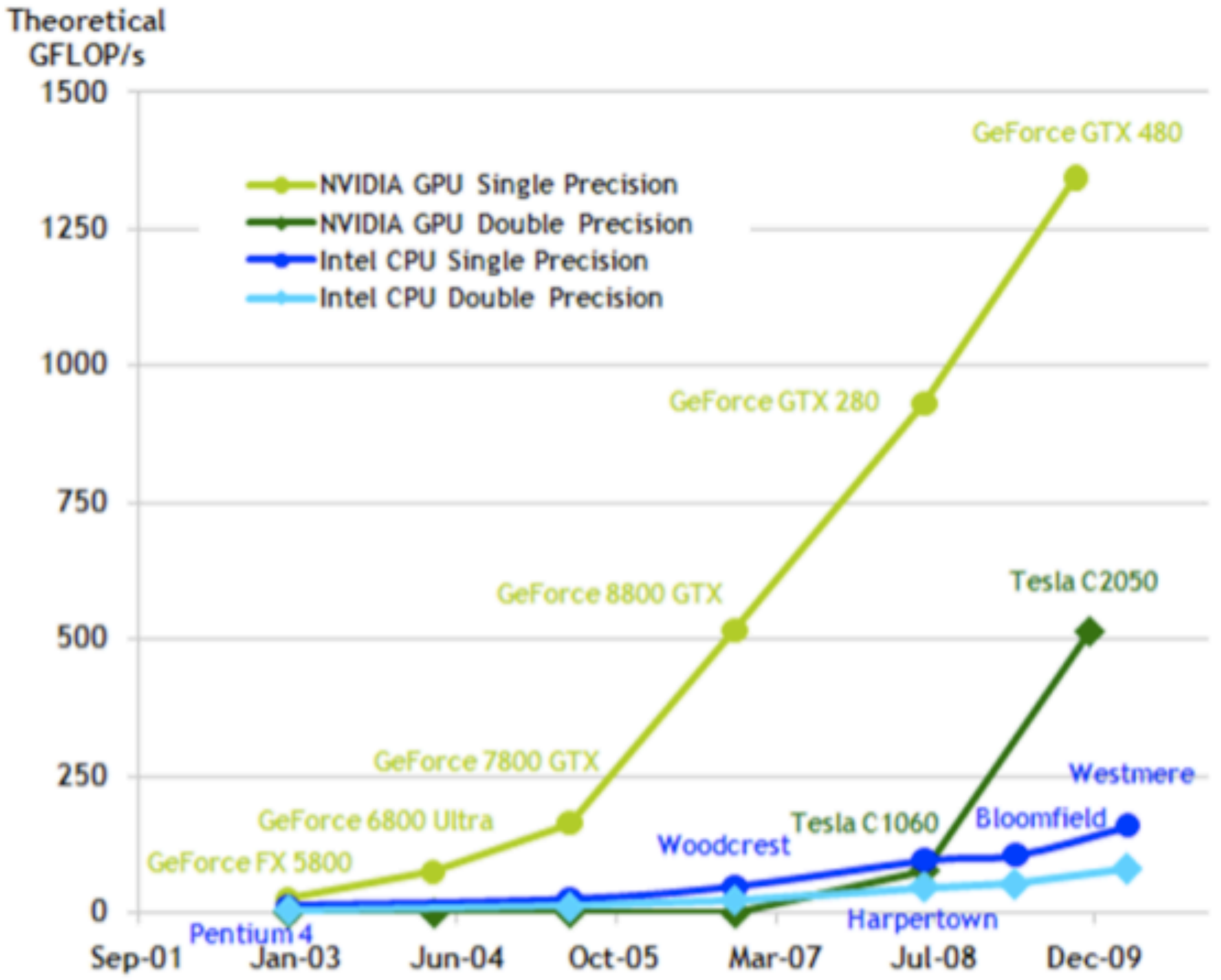
Geophysics

Biophysics



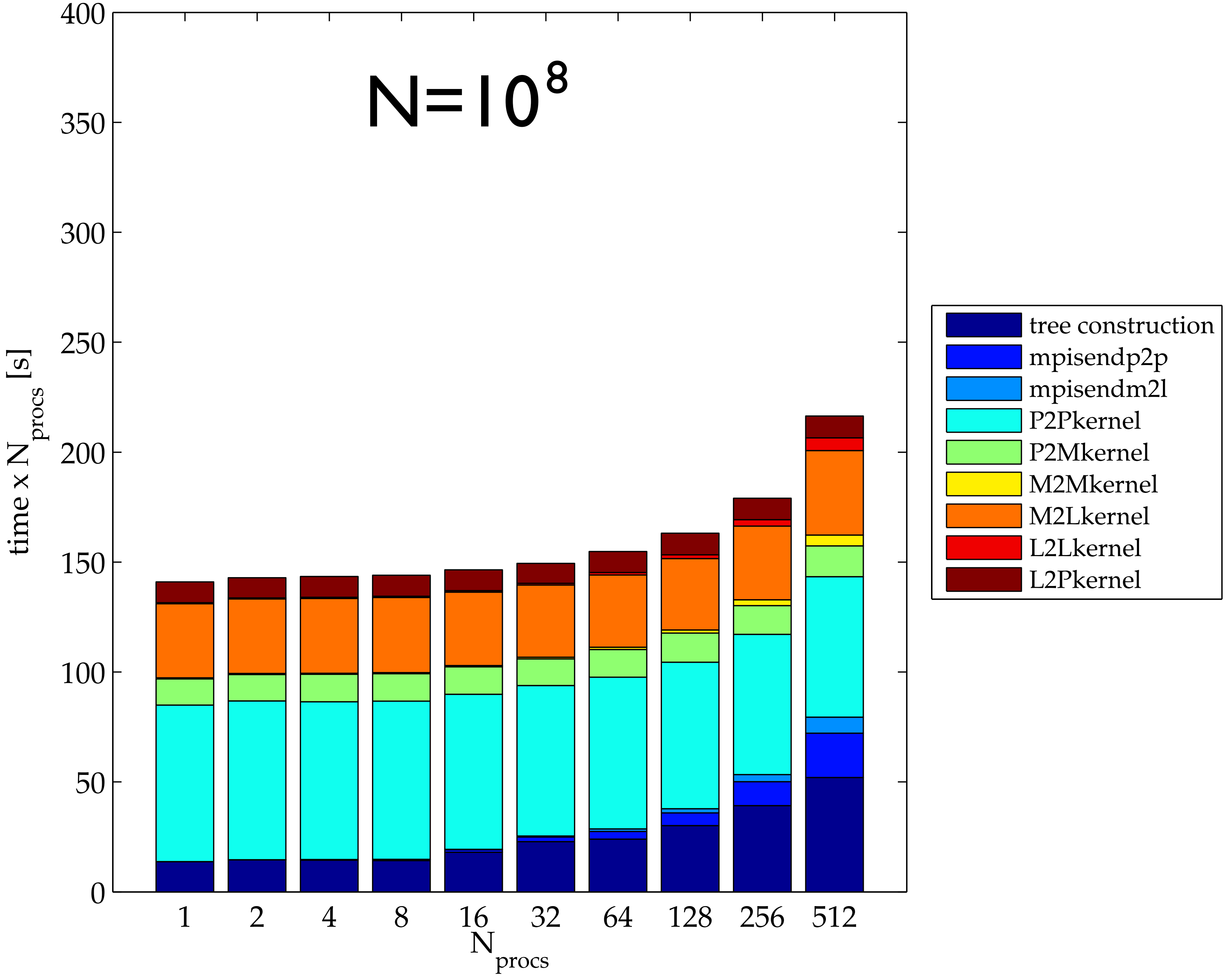
Scalability on many GPUs

Rank	Site	Computer/Year Vendor	Cores	R_{max}	R_{peak}	Power
1	National Supercomputing Center in Tianjin China	Tianhe-1A - NUDT YH Cluster, X5670 2.93Ghz 6C, NVIDIA GPU, FT-1000 8C / 2010 NUDT	186368	2566.00	4701.00	4040.00
2	DOE/SC/Oak Ridge National Laboratory United States	Jaguar - Cray XT5-HE Opteron 6-core 2.6 GHz / 2009 Cray Inc.	224162	1759.00	2331.00	6950.60
3	National Supercomputing Centre in Shenzhen (NSCS) China	Nebulae - Dawning TC3600 Blade, Intel X5650, NVidia Tesla C2050 GPU / 2010 Dawning	120640	1271.00	2984.30	2580.00
4	GSIC Center, Tokyo Institute of Technology Japan	TSUBAME 2.0 - HP ProLiant SL390s G7 Xeon 6C X5670, Nvidia GPU, Linux/Windows / 2010 NEC/HP	73278	1192.00	2287.63	1398.61
5	DOE/SC/LBNL/NERSC United States	Hopper - Cray XE6 12-core 2.1 GHz / 2010 Cray Inc.	153408	1054.00	1288.63	2910.00

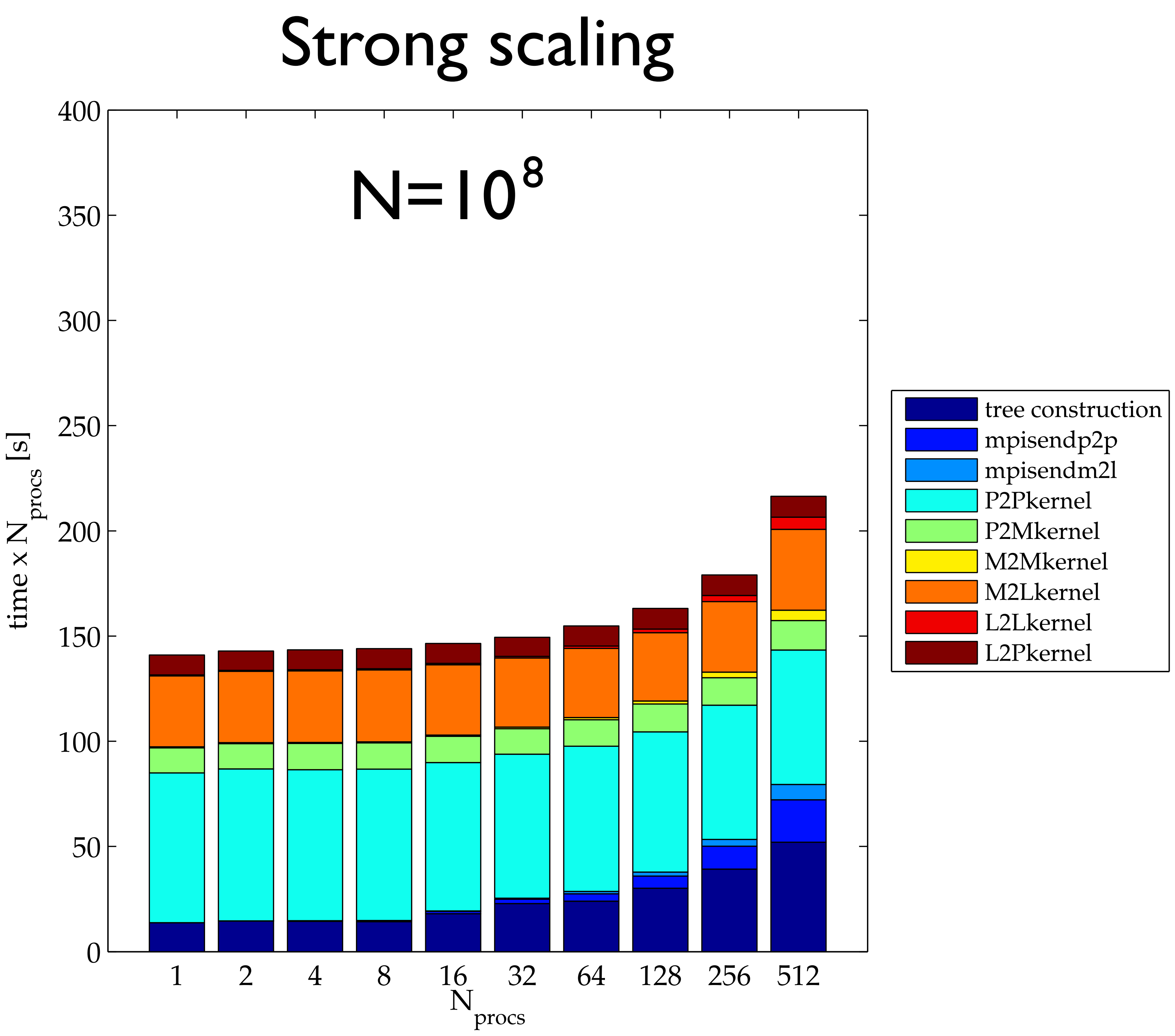


Scalability on many GPUs

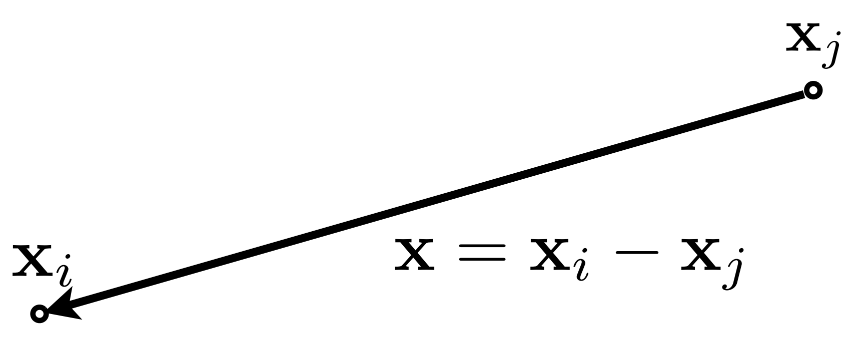
Strong scaling



Scalability on many GPUs



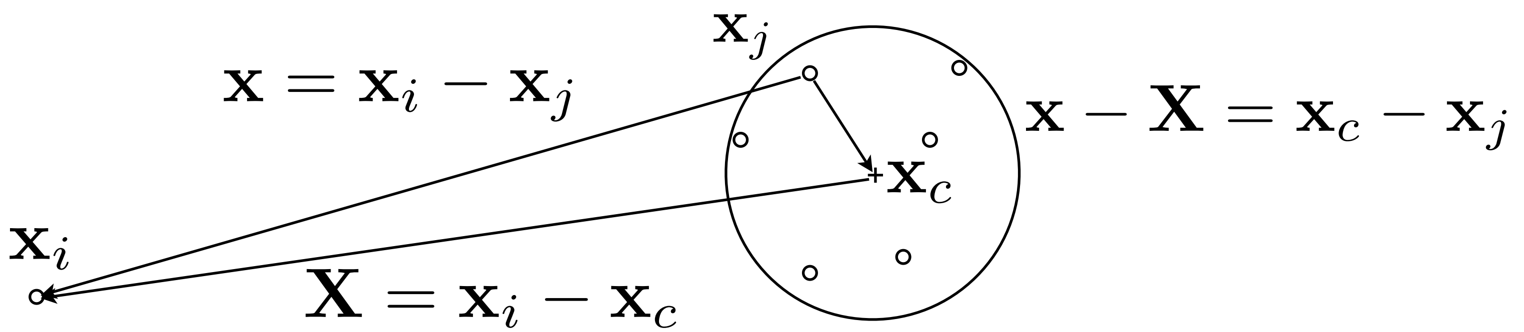
N-body interactions



$$\Phi_i = \sum_{j=0}^{N} \frac{m_j}{r}$$

$$\begin{aligned} r &= \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \\ &= \sqrt{x^2 + y^2 + z^2} \\ &= |\mathbf{x}| \end{aligned}$$

Taylor expansion



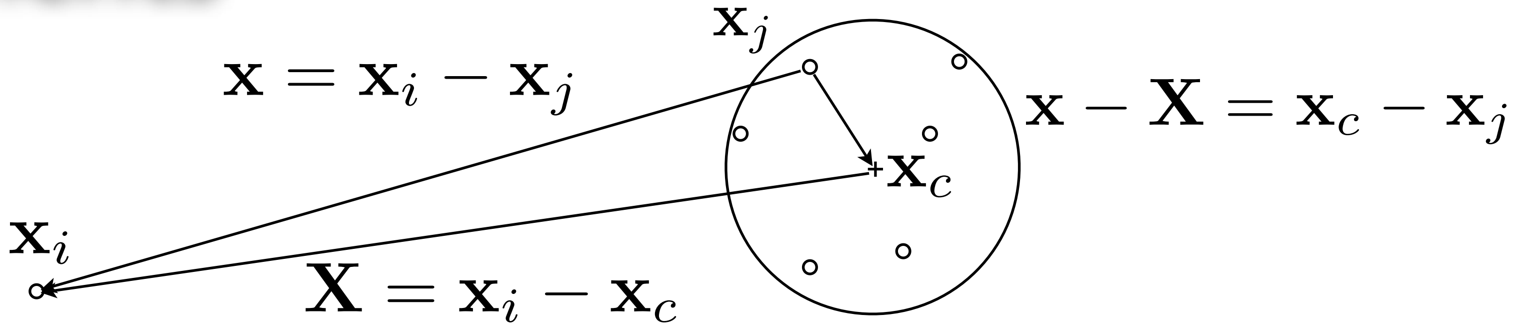
$$\Phi_i = \sum_{j=0}^{N} m_j f(\mathbf{x})$$

$$f(\mathbf{x}) = f(\mathbf{X}) + (\mathbf{x} - \mathbf{X})\frac{f'(\mathbf{X})}{1!} + (\mathbf{x} - \mathbf{X})^2 \frac{f''(\mathbf{X})}{2!} + \ldots$$

$$R = \sqrt{X^2 + Y^2 + Z^2}$$

$$\begin{aligned}
\frac{1}{r} = \frac{1}{R} &+ \frac{1}{1!}\left[(x - X)\frac{\partial}{\partial X}\frac{1}{R} + (y - Y)\frac{\partial}{\partial Y}\frac{1}{R} + (z - Z)\frac{\partial}{\partial Z}\right] \\
&+ \frac{1}{2!}\left[(x - X)^2\frac{\partial^2}{\partial X \partial X}\frac{1}{R} + (y - Y)^2\frac{\partial^2}{\partial Y \partial Y}\frac{1}{R} + (z - Z)^2\frac{\partial^2}{\partial Z \partial Z}\frac{1}{R}\right. \\
&\left. + (x - X)(y - Y)\frac{\partial^2}{\partial X \partial Y}\frac{1}{R} + (y - Y)(z - Z)\frac{\partial^2}{\partial Y \partial Z}\frac{1}{R} + (z - Z)(x - X)\frac{\partial^2}{\partial Z \partial X}\frac{1}{R}\right]
\end{aligned}$$

Taylor expansion :: components



$$\frac{\partial}{\partial X}\frac{1}{R} = -\frac{X}{R^3}$$

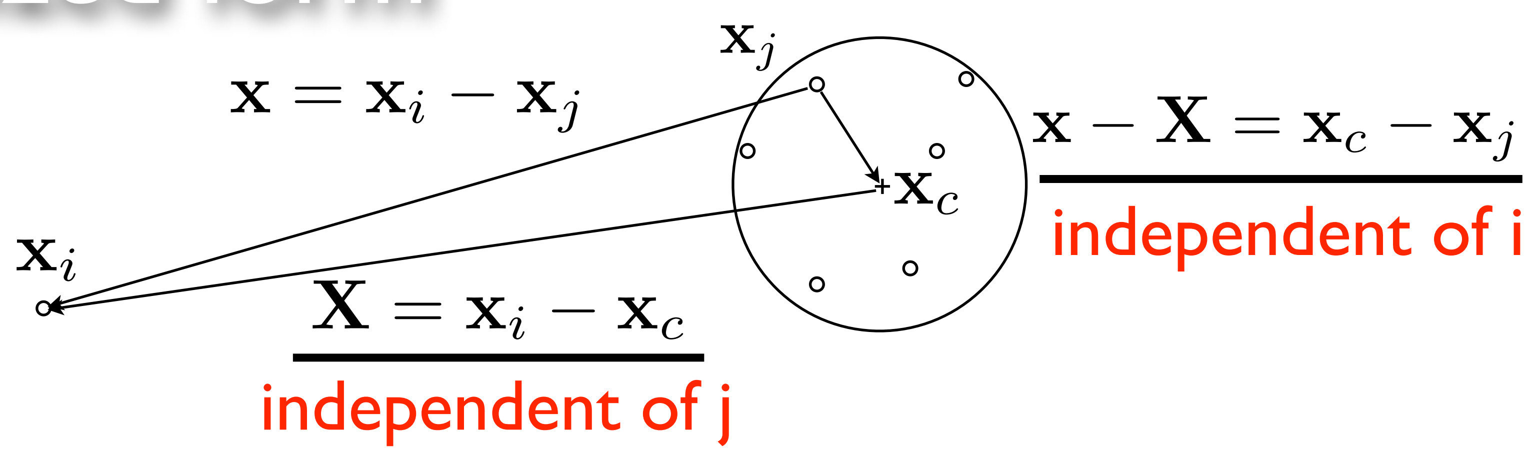
$$\frac{\partial^2}{\partial X \partial X}\frac{1}{R} = \frac{3X^2}{R^5} - \frac{1}{R^3}$$

$$\frac{\partial^2}{\partial X \partial Y}\frac{1}{R} = \frac{3XY}{R^5}$$

$$\begin{aligned}\frac{1}{r} = \frac{1}{R} &+ \frac{1}{1!}\left[(x-X)\left(-\frac{X}{R^3}\right) + (y-Y)\left(-\frac{Y}{R^3}\right) + (z-Z)\left(-\frac{Z}{R^3}\right)\right] \\ &+ \frac{1}{2!}\left[(x-X)^2\left(\frac{3X^2}{R^5} - \frac{1}{R^3}\right) + (y-Y)^2\left(\frac{3Y^2}{R^5} - \frac{1}{R^3}\right) + (z-Z)^2\left(\frac{3Z^2}{R^5} - \frac{1}{R^3}\right)\right. \\ &\left. + (x-X)(y-Y)\frac{3XY}{R^5} + (y-Y)(z-Z)\frac{3YZ}{R^5} + (z-Z)(x-X)\frac{3ZX}{R^5}\right]\end{aligned}$$

Taylor expansion :: generalized form

$$\frac{1}{r} = \sum_{n=0}^{p} \frac{1}{n!} (\mathbf{x} - \mathbf{X})^n \frac{\partial^{(n)}}{\partial \mathbf{X}} \frac{1}{R}$$



$$\Phi_i = \sum_{j=0}^{N} \frac{m_j}{r} = \sum_{j=0}^{N} m_j \sum_{n=0}^{p} \frac{1}{n!} (\mathbf{x} - \mathbf{X})^n \frac{\partial^{(n)}}{\partial \mathbf{X}} \frac{1}{R}$$

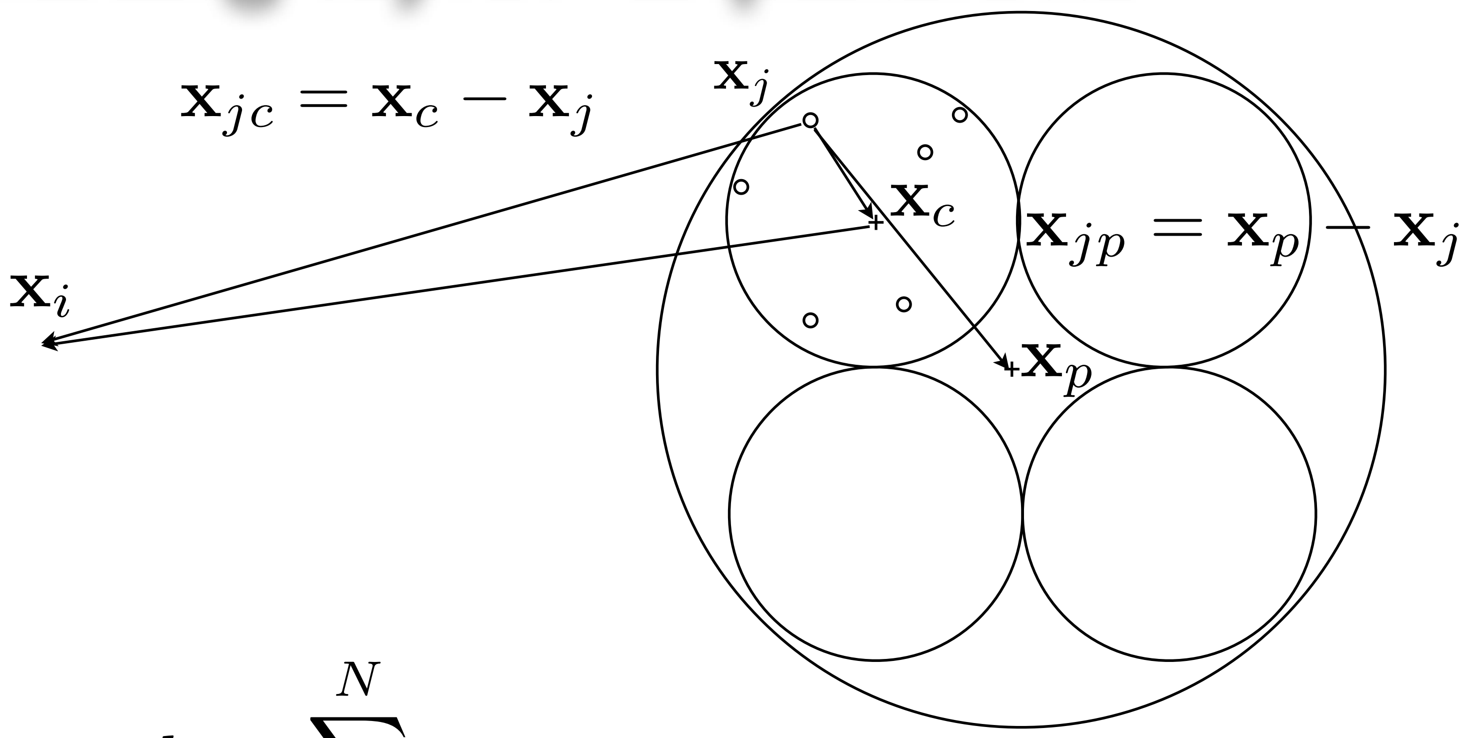
$$= \sum_{n=0}^{p} \frac{\partial^{(n)}}{\partial \mathbf{X}} \frac{1}{R} \underbrace{\sum_{j=0}^{N} \frac{1}{n!} m_j (\mathbf{x} - \mathbf{X})^n}_{Multipole}$$

$$\frac{1}{R^{n+1}} \qquad\qquad (r - R)^n$$

$$\left(\frac{r - R}{R} \right)^n < 1$$

$$|\mathbf{x} - \mathbf{X}| < |\mathbf{X}|$$

Shifting Taylor expansions



$$monopole = \sum_{j=0}^{N} m_j$$

$$dipole = \sum_{j=0}^{N} m_j \mathbf{x}_{jc}$$

$$quadrupole = \sum_{j=0}^{N} \frac{1}{2} m_j \mathbf{x}_{jc}^2$$

$$monopole = \sum_{j=0}^{N} m_j$$

$$dipole = \sum_{j=0}^{N} m_j \mathbf{x}_{jp} = \sum_{j=0}^{N} m_j (\mathbf{x}_{jc} + \mathbf{x}_{cp}) = \underbrace{\sum_{j=0}^{N} m_j \mathbf{x}_{jc}}_{dipole} + \mathbf{x}_{cp} \underbrace{\sum_{j=0}^{N} m_j}_{monopole}$$

$$quadrupole = \sum_{j=0}^{N} \frac{1}{2} m_j \mathbf{x}_{jp}^2 = \sum_{j=0}^{N} \frac{1}{2} m_j (\mathbf{x}_{jc} + \mathbf{x}_{cp})^2 = \underbrace{\sum_{j=0}^{N} \frac{1}{2} m_j \mathbf{x}_{jc}^2}_{quadrupole} + \mathbf{x}_{cp} \underbrace{\sum_{j=0}^{N} m_j \mathbf{x}_{jc}}_{dipole} + \frac{\mathbf{x}_{cp}^2}{2} \underbrace{\sum_{j=0}^{N} m_j}_{monopole}$$

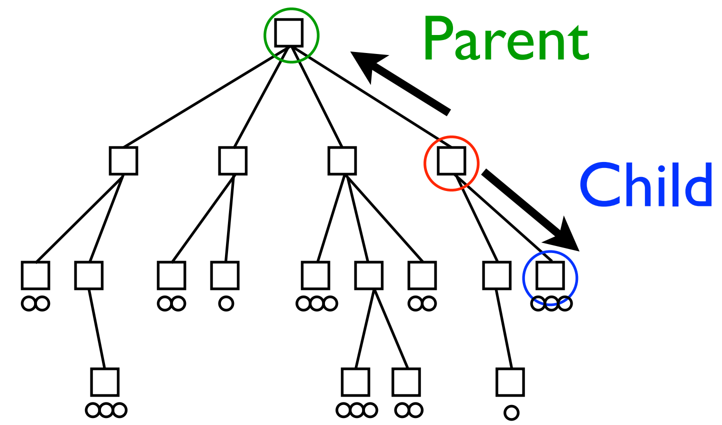
Shifting Taylor expansions :: components

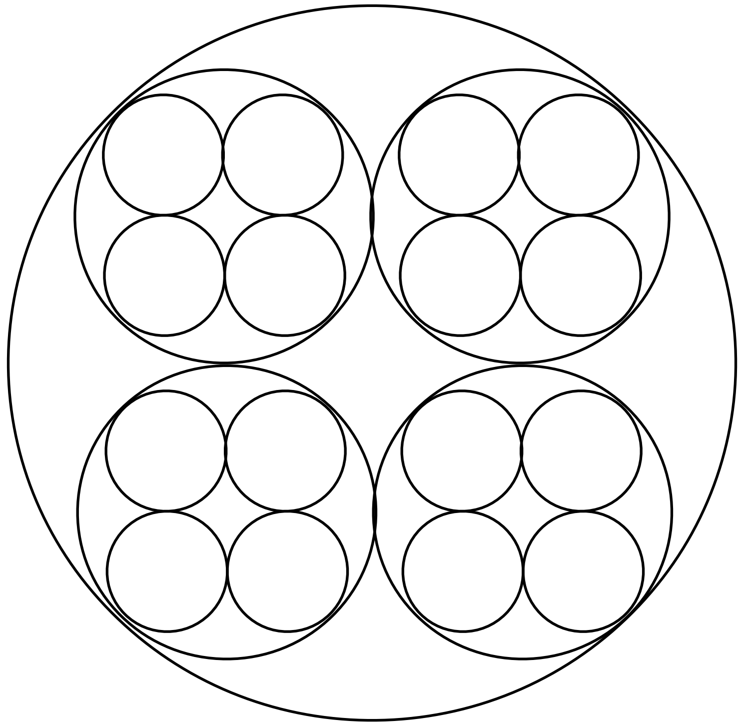
$$quadrupole = \sum_{j=0}^{N} \frac{1}{2} m_j \mathbf{x}_{jp}^2 = \sum_{j=0}^{N} \frac{1}{2} m_j (\mathbf{x}_{jc} + \mathbf{x}_{cp})^2 = \underbrace{\sum_{j=0}^{N} \frac{1}{2} m_j \mathbf{x}_{jc}^2}_{quadrupole} + \mathbf{x}_{cp} \underbrace{\sum_{j=0}^{N} m_j \mathbf{x}_{jc}}_{dipole} + \frac{\mathbf{x}_{cp}^2}{2} \underbrace{\sum_{j=0}^{N} m_j}_{monopole}$$

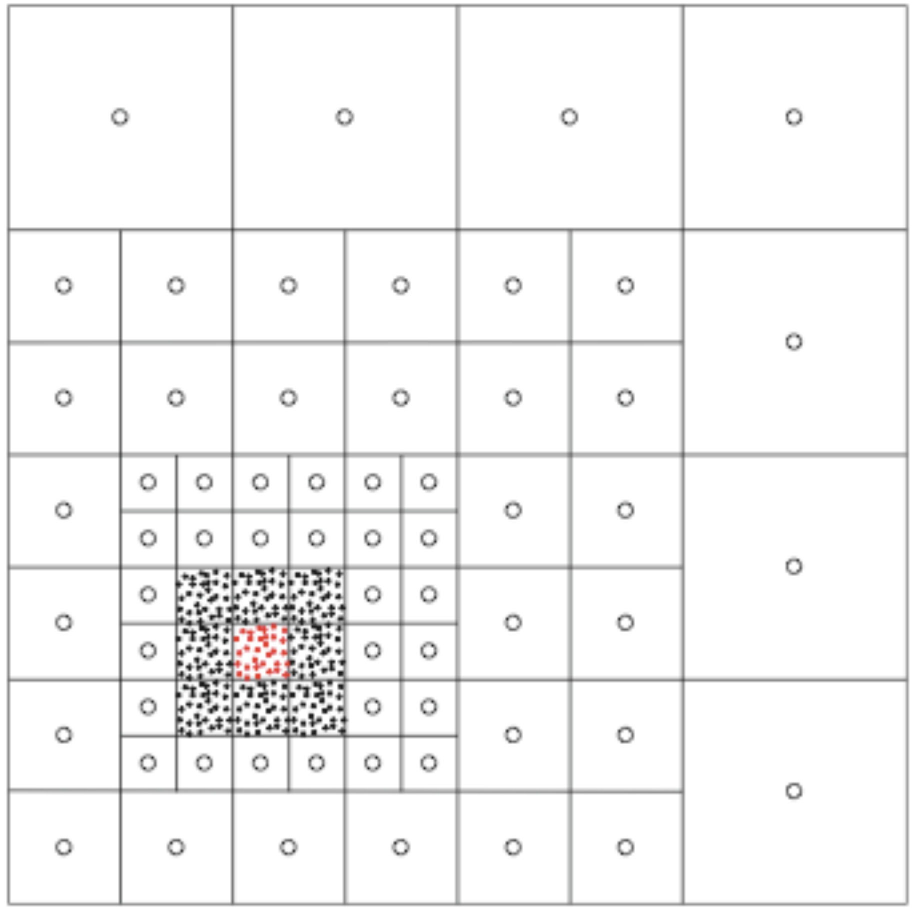
$$\sum_{j=0}^{N} \frac{1}{2} m_j x_{jp}^2 = \sum_{j=0}^{N} \frac{1}{2} m_j x_{jc}^2 + x_{cp} \sum_{j=0}^{N} m_j x_{jc} + \frac{x_{cp}^2}{2} \sum_{j=0}^{N} m_j$$

$$\sum_{j=0}^{N} \frac{1}{2} m_j x_{jp} y_{jp} = \frac{1}{2} \left(\sum_{j=0}^{N} m_j x_{jc} y_{jc} + x_{cp} \sum_{j=0}^{N} m_j y_{jc} + y_{cp} \sum_{j=0}^{N} m_j x_{jc} + x_{cp} y_{cp} \sum_{j=0}^{N} m_j \right)$$

Tree structure



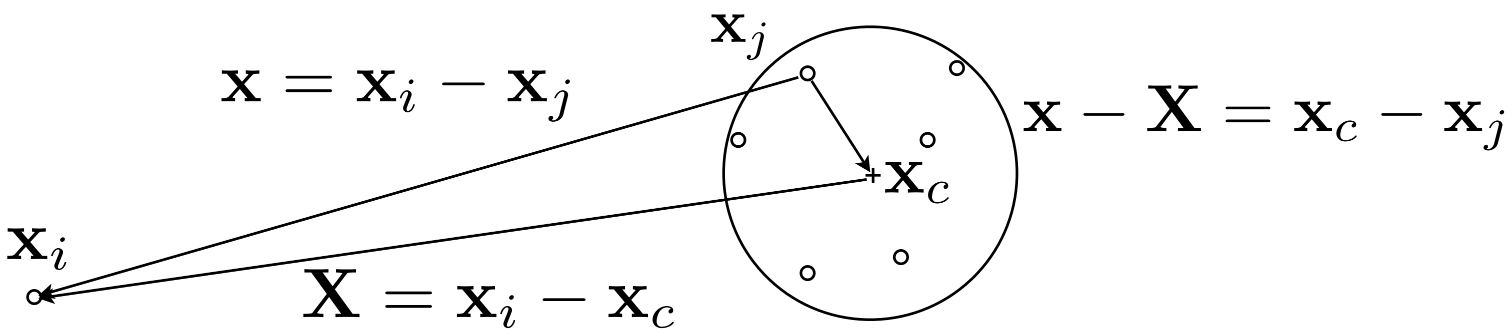




$$\Phi_i = \sum_{j=0}^{N} \frac{m_j}{r} = \sum_{j=0}^{N} m_j \sum_{n=0}^{p} \frac{1}{n!} (\mathbf{x} - \mathbf{X})^n \frac{\partial^{(n)}}{\partial \mathbf{X}} \frac{1}{R}$$

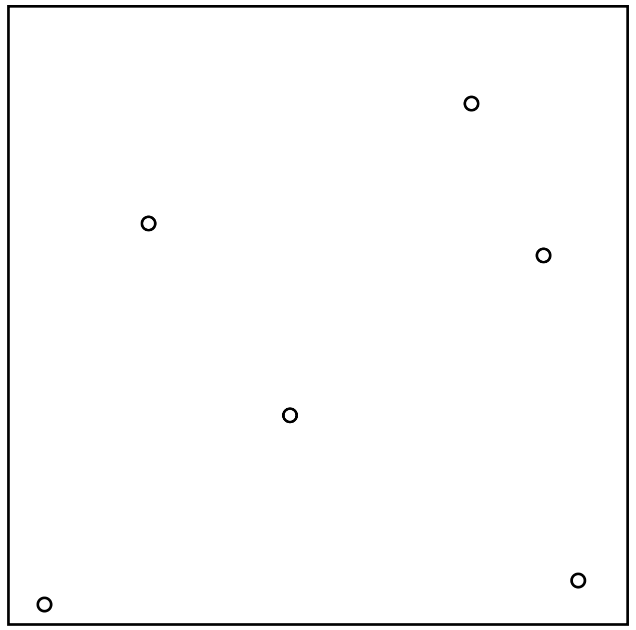
$$= \sum_{n=0}^{p} \frac{\partial^{(n)}}{\partial \mathbf{X}} \frac{1}{R} \underbrace{\sum_{j=0}^{N} \frac{1}{n!} m_j (\mathbf{x} - \mathbf{X})^n}_{Multipole}$$

$$|\mathbf{x} - \mathbf{X}| < |\mathbf{X}|$$

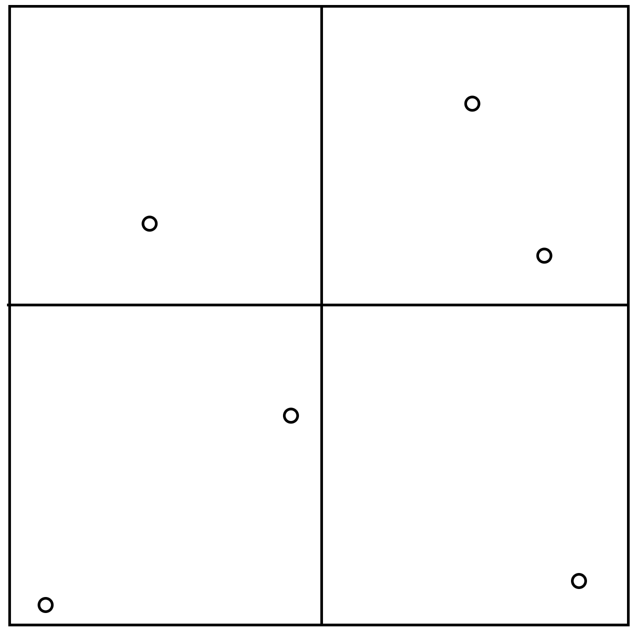


Tree structure :: constructing the tree

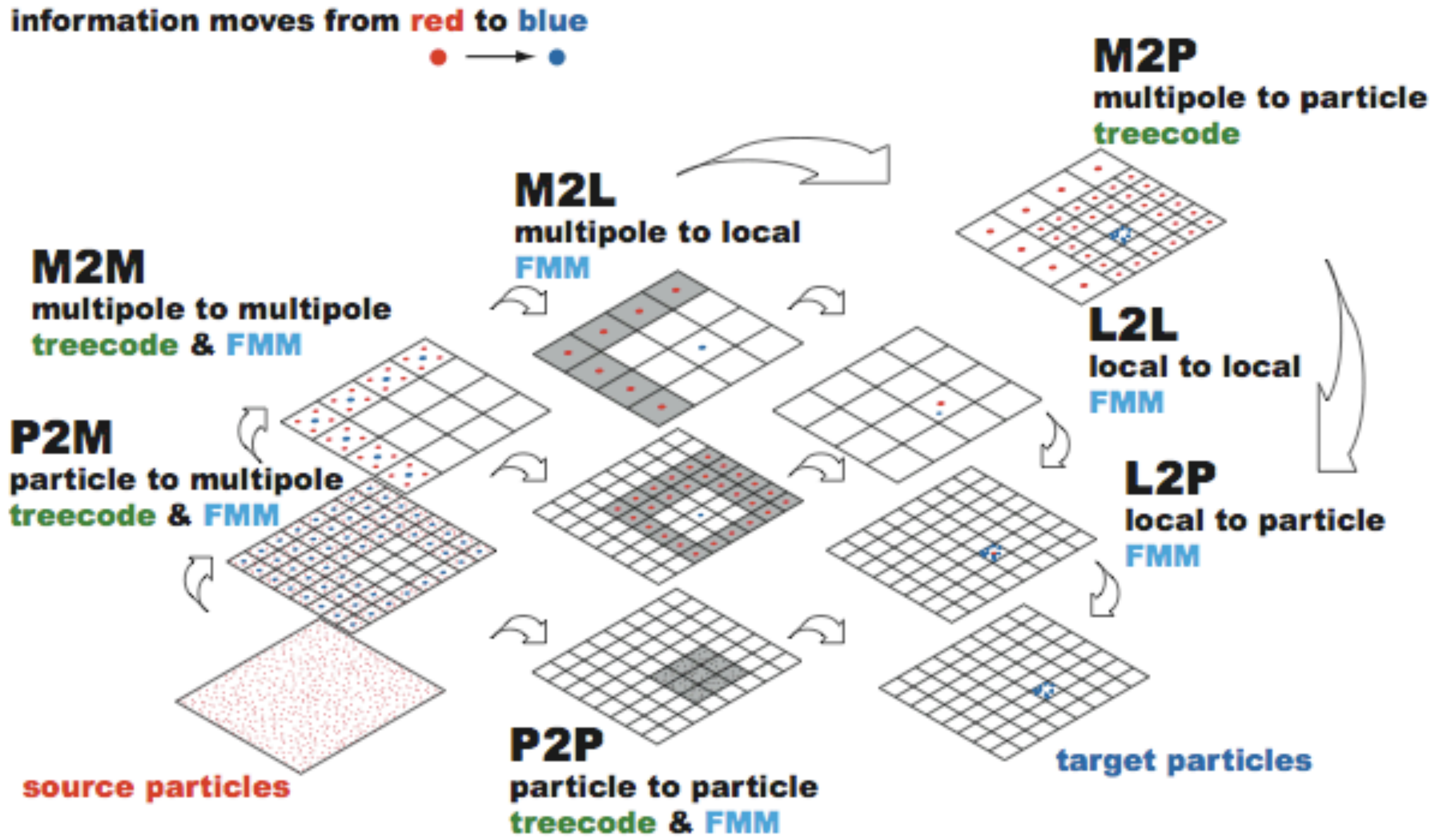
Step a. Keep putting particles into root cell

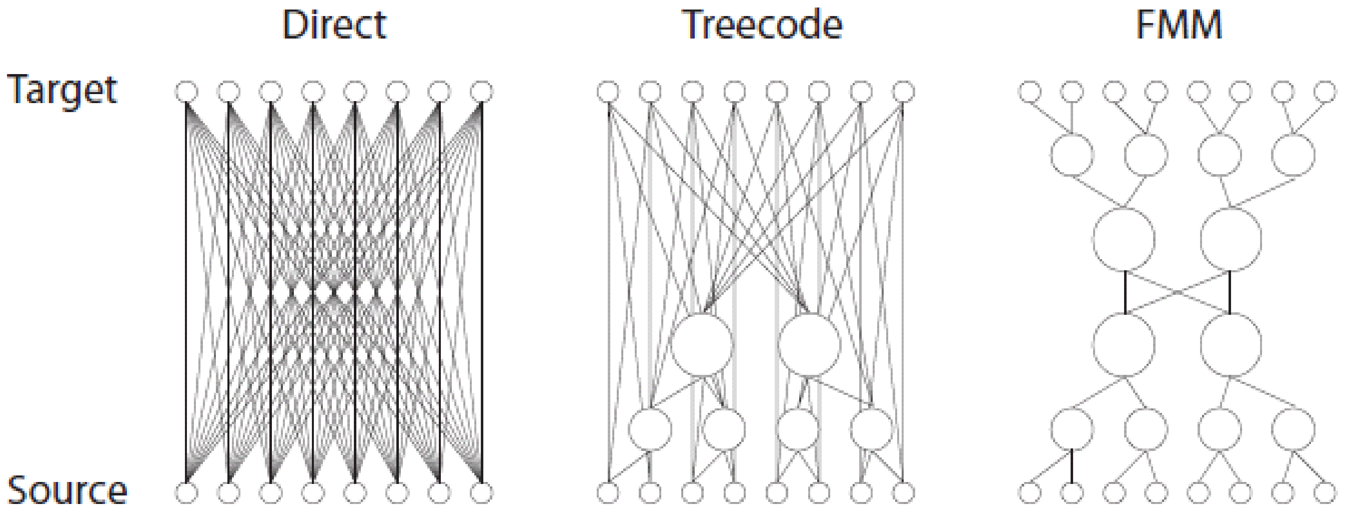


Step b. If number of particles becomes larger than a threshold, subdivide cells



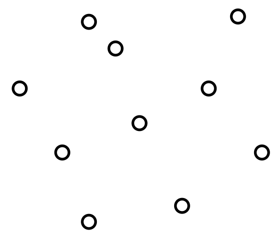
Fast Multipole Methods





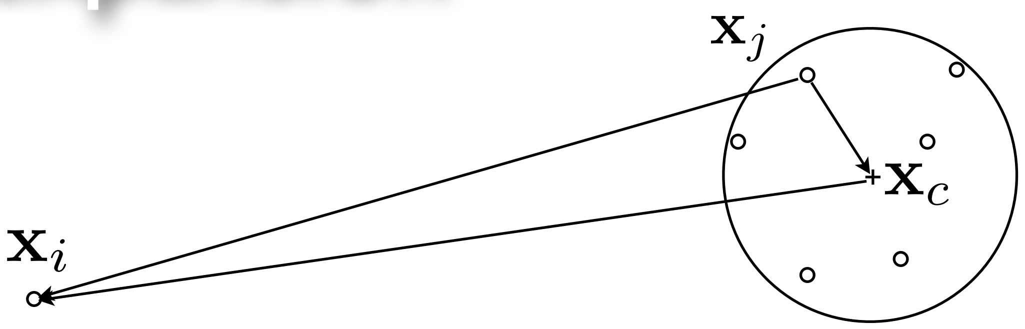
Step01. direct N-body summation

$$\Phi_i = \sum_{\substack{j=0 \\ i \neq j}}^{N} \frac{m_j}{r} = \sum_{\substack{j=0 \\ i \neq j}}^{N} \frac{m_j}{\sqrt{r^2 + \epsilon^2}}$$



```
int main() {
  int N = 10;
  float x[N],y[N],z[N],m[N];
// Initialize
  for( int i=0; i<N; i++ ) {
    x[i] = rand()/(1.+RAND_MAX);
    y[i] = rand()/(1.+RAND_MAX);
    z[i] = rand()/(1.+RAND_MAX);
    m[i] = 1.0/N;
  }
// Direct summation
  float dx,dy,dz,r,eps2=0.0001;
  for( int i=0; i<N; i++ ) {
    float p = -m[i]/sqrtf(eps2);
    for( int j=0; j<N; j++ ) {
      dx = x[i]-x[j];
      dy = y[i]-y[j];
      dz = z[i]-z[j];
      r = sqrtf(dx*dx+dy*dy+dz*dz+eps2);
      p += m[j] / r;
    }
  }
}
```

Step02. multipole expansion



$$\Phi_i = \sum_{j=0}^{N} \frac{m_j}{r} = \sum_{j=0}^{N} m_j \sum_{n=0}^{p} \frac{1}{n!} (\mathbf{x} - \mathbf{X})^n \frac{\partial^{(n)}}{\partial \mathbf{X}} \frac{1}{R}$$

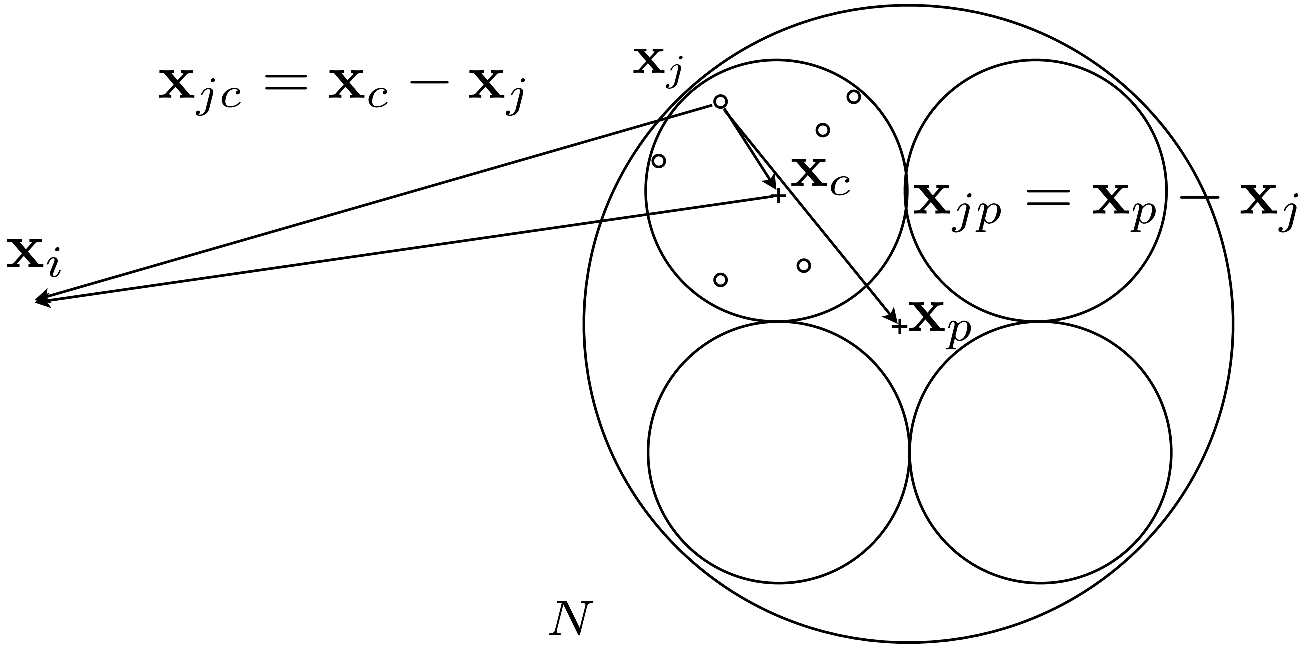
$$= \sum_{n=0}^{p} \frac{\partial^{(n)}}{\partial \mathbf{X}} \frac{1}{R} \underbrace{\sum_{j=0}^{N} \frac{1}{n!} m_j (\mathbf{x} - \mathbf{X})^n}_{Multipole}$$

$$\frac{1}{r} = \frac{1}{R} + \frac{1}{1!} \left[(x - X) \left(-\frac{X}{R^3} \right) + (y - Y) \left(-\frac{Y}{R^3} \right) + (z - Z) \left(-\frac{Z}{R^3} \right) \right]$$
$$+ \frac{1}{2!} \left[(x - X)^2 \left(\frac{3X^2}{R^5} - \frac{1}{R^3} \right) + (y - Y)^2 \left(\frac{3Y^2}{R^5} - \frac{1}{R^3} \right) + (z - Z)^2 \left(\frac{3Z^2}{R^5} - \frac{1}{R^3} \right) \right.$$
$$\left. + (x - X)(y - Y) \frac{3XY}{R^5} + (y - Y)(z - Z) \frac{3YZ}{R^5} + (z - Z)(x - X) \frac{3ZX}{R^5} \right]$$

```
// Multipole expansion
  float xc=0.5, yc=0.5, zc=0.5;
  float multipole[10]={0,0,0,0,0,0,0,0,0,0};
  for( int j=0; j<N; j++ ) {
    dx = xc-xj[j];
    dy = yc-yj[j];
    dz = zc-zj[j];
    multipole[0] += mj[j];
    multipole[1] += mj[j]*dx;
    multipole[2] += mj[j]*dy;
    multipole[3] += mj[j]*dz;
    multipole[4] += mj[j]*dx*dx/2;
    multipole[5] += mj[j]*dy*dy/2;
    multipole[6] += mj[j]*dz*dz/2;
    multipole[7] += mj[j]*dx*dy/2;
    multipole[8] += mj[j]*dy*dz/2;
    multipole[9] += mj[j]*dz*dx/2;
  }

 // Evaluate potential
  float X,Y,Z,R,R3,R5;
  for( int i=0; i<N; i++ ) {
    float p = 0;
    X = xi[i]-xc;
    Y = yi[i]-yc;
    Z = zi[i]-zc;
    R = sqrtf(X*X+Y*Y+Z*Z);
    R3 = R*R*R;
    R5 = R3*R*R;
    p += multipole[0]/R;
    p += multipole[1]*(-X/R3);
    p += multipole[2]*(-Y/R3);
    p += multipole[3]*(-Z/R3);
    p += multipole[4]*(3*X*X/R5-1/R3);
    p += multipole[5]*(3*Y*Y/R5-1/R3);
    p += multipole[6]*(3*Z*Z/R5-1/R3);
    p += multipole[7]*(3*X*Y/R5);
    p += multipole[8]*(3*Y*Z/R5);
    p += multipole[9]*(3*Z*X/R5);
  }
```

Step03. multipole expansion (multi-level)




```
// Upward translation
  for( int i=0; i<8; i++ ) {
    dx = xp-xc[i];
    dy = yp-yc[i];
    dz = zp-zc[i];
    multipole[8][0] += multipole[i][0];
    multipole[8][1] += multipole[i][1]+ dx*multipole[i][0];
    multipole[8][2] += multipole[i][2]+ dy*multipole[i][0];
    multipole[8][3] += multipole[i][3]+ dz*multipole[i][0];
    multipole[8][4] += multipole[i][4]+ dx*multipole[i][1]+dx*dx*multipole[i][0]/2;
    multipole[8][5] += multipole[i][5]+ dy*multipole[i][2]+dy*dy*multipole[i][0]/2;
    multipole[8][6] += multipole[i][6]+ dz*multipole[i][3]+dz*dz*multipole[i][0]/2;
    multipole[8][7] += multipole[i][7]+(dx*multipole[i][2]+dy*multipole[i][1]+dx*dy*multipole[i][0])/2;
    multipole[8][8] += multipole[i][8]+(dy*multipole[i][3]+dz*multipole[i][2]+dy*dz*multipole[i][0])/2;
    multipole[8][9] += multipole[i][9]+(dz*multipole[i][1]+dx*multipole[i][3]+dz*dx*multipole[i][0])/2;
  }
```

$$monopole = \sum_{j=0}^{N} m_j$$

$$dipole = \sum_{j=0}^{N} m_j \mathbf{x}_{jp} = \sum_{j=0}^{N} m_j (\mathbf{x}_{jc} + \mathbf{x}_{cp}) = \underbrace{\sum_{j=0}^{N} m_j \mathbf{x}_{jc}}_{dipole} + \mathbf{x}_{cp} \underbrace{\sum_{j=0}^{N} m_j}_{monopole}$$

$$quadrupole = \sum_{j=0}^{N} \frac{1}{2} m_j \mathbf{x}_{jp}^2 = \sum_{j=0}^{N} \frac{1}{2} m_j (\mathbf{x}_{jc} + \mathbf{x}_{cp})^2 = \underbrace{\sum_{j=0}^{N} \frac{1}{2} m_j \mathbf{x}_{jc}^2}_{quadrupole} + \mathbf{x}_{cp} \underbrace{\sum_{j=0}^{N} m_j \mathbf{x}_{jc}}_{dipole} + \frac{\mathbf{x}_{cp}^2}{2} \underbrace{\sum_{j=0}^{N} m_j}_{monopole}$$