

# FMM CMSC 878R/AMSC 698R

## Lecture 10

# Decomposition of the matrix vector product

$$\Phi_2^{(n)}(\mathbf{y}) = \sum_{\mathbf{x}_i \in E_2(n)} u_i \Phi(\mathbf{y}, \mathbf{x}_i),$$

$$\Phi_3^{(n)}(\mathbf{y}) = \sum_{\mathbf{x}_i \in E_3(n)} u_i \Phi(\mathbf{y}, \mathbf{x}_i),$$

Since domains  $E_2(n)$  and  $E_3(n)$  are complimentary:

$$\Phi(\mathbf{y}) = \sum_{i=1}^N u_i \Phi(\mathbf{y}, \mathbf{x}_i) = \sum_{\mathbf{x}_i \in E_2(n) \cup E_3(n)} u_i \Phi(\mathbf{y}, \mathbf{x}_i) = \Phi_2^{(n)}(\mathbf{y}) + \Phi_3^{(n)}(\mathbf{y})$$

for arbitrary  $n$ .

- FMM and related algorithms are about the approximation of the product  $\Phi_3$

# SLFMM Algorithm

Step 1. Generate S-expansion coefficients  
for each box

$$\Phi_1^{(n)}(\mathbf{x}) = \mathbf{C}^{(n)} \circ \mathbf{S}(\mathbf{x} - \mathbf{x}_c^{(n)}),$$
$$\mathbf{C}^{(n)} = \sum_{\mathbf{x}_i \in E_1(n,L)} u_i \mathbf{B}(\mathbf{x}_i, \mathbf{x}_c^{(n)}).$$

loop over all non-empty source boxes

*For*  $n \in \text{NonEmptySource}$

Get  $\mathbf{x}_c^{(n)}$ , the center of the box;

$\mathbf{C}^{(n)} = \mathbf{0}$ ;

*For*  $\mathbf{x}_i \in E_1(n)$

loop over all sources in the box

Get  $\mathbf{B}(\mathbf{x}_i, \mathbf{x}_c^{(n)})$ , the S-expansion coefficients  
near the center of the box;

$\mathbf{C}^{(n)} = \mathbf{C}^{(n)} + u_i \mathbf{B}(\mathbf{x}_i, \mathbf{x}_c^{(n)})$ ;

*End*;

*End*;

**Implementation can be different!**

**All we need is to get  $\mathbf{C}^{(n)}$ .**

# SLFMM Algorithm

Step 2. (S|R)-translate expansion coefficients

$$\Phi_3^{(n)}(\mathbf{y}) = \mathbf{D}^{(n)} \circ \mathbf{R}(\mathbf{y} - \mathbf{x}_c^{(n)}),$$
$$\mathbf{D}^{(n)} = \sum_{m \in I_3(n)} (\mathbf{S|R})(\mathbf{x}_c^{(n)} - \mathbf{x}_c^{(m)}) \mathbf{C}^{(m)}.$$

loop over all non-empty  
evaluation boxes

*For*  $n \in \text{NonEmptyEvaluation}$

Get  $\mathbf{x}_c^{(n)}$ , the center of the box;

$\mathbf{D}^{(n)} = \mathbf{0}$ ;

loop over all non-empty source boxes

*For*  $m \in I_3(n)$  ← outside the neighborhood of the  $n$ -th box

Get  $\mathbf{x}_c^{(m)}$ , the center of the box;

$\mathbf{D}^{(n)} = \mathbf{D}^{(n)} + (\mathbf{S|R})(\mathbf{x}_c^{(n)} - \mathbf{x}_c^{(m)}) \mathbf{C}^{(m)}$ ;

*End*;

*End*;


**Implementation can be different!**

**All we need is to get  $\mathbf{D}^{(n)}$ .**


# SLFMM Algorithm

## Step 3. Final Summation


$$v_j = \Phi(\mathbf{y}_j) = \sum_{\mathbf{x}_i \in E_2(n)} u_i \Phi(\mathbf{y}_j, \mathbf{x}_i) + \mathbf{D}^{(n)} \circ \mathbf{R}(\mathbf{y}_j - \mathbf{x}_c^{(n)}), \quad \mathbf{y}_j \in E_1(n).$$

*For*  $n \in \text{NonEmptyEvaluation}$   loop over all boxes  
containing evaluation points

Get  $\mathbf{x}_c^{(n)}$ , the center of the box;

*For*  $\mathbf{y}_j \in E_1(n)$   loop over all evaluation points in the box

$v_j = \mathbf{D}^{(n)} \circ \mathbf{R}(\mathbf{y}_j - \mathbf{x}_c^{(n)})$ ;

*For*  $\mathbf{x}_i \in E_2(n)$   loop over all sources in the  
neighborhood of the  $n$ -th box

$v_j = v_j + \Phi(\mathbf{y}_j, \mathbf{x}_i) * u_i$ ;

*End;*

*End;*

*End;*

# Asymptotic Complexity of SLFMM

- By some magic we can easily find neighbors, and lists of points in each box.
  - Assume that cost of finding these is less than the most expensive operation
  - Magic will come from data-structures
- Translation is performed by straightforward  $P \times P$  matrix-vector multiplication, where  $P(p)$  is the total length of the translation vector. So the complexity of a single translation is  $O(P^2)$ .
- The source and evaluation points are distributed uniformly, and there are  $K$  boxes, with  $s$  source points in each box ( $s=N/K$ ). We call  $s$  the *grouping* (or *clustering*) parameter.
- The number of neighbors for each box is  $O(1)$ .

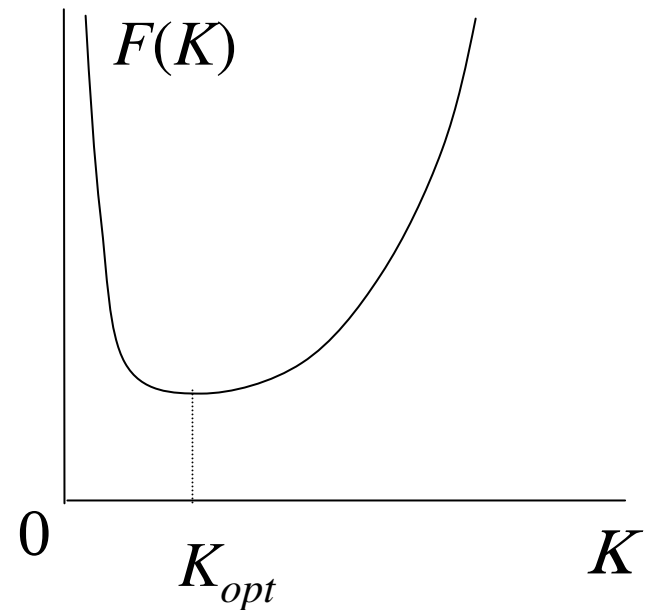
# Then Complexity is:

- For Step 1:  $O(PN)$
- For Step 2:  $O(P^2K^2)$
- For Step 3:  $O(PM+Ms)$
- Total:  $O(PN+ P^2K^2 +PM+Ms) =$   
 $O(PN+ P^2K^2 +PM+MN/K)$

# Selection of Optimal $K$ (or $s$ )

$$F(K) = PN + P^2K^2 + PM + MN/K.$$

$$F'(K) = 2P^2K - MN/K^2 = 0.$$



$$K_{opt} = ((MN) / (2P^2))^{1/3}$$

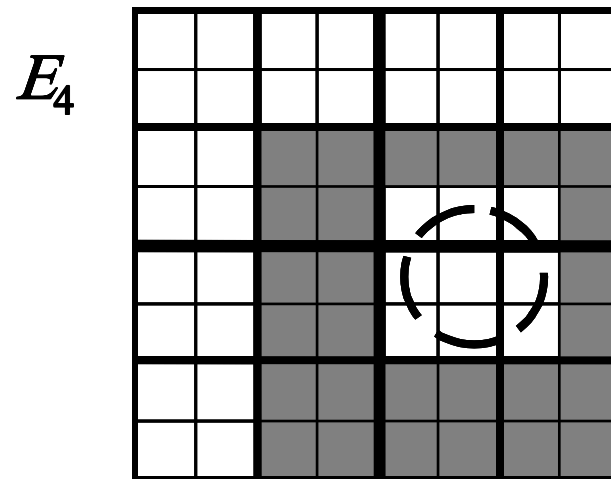
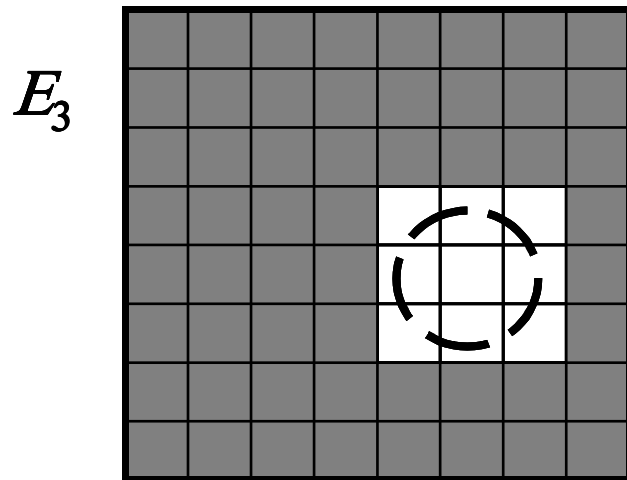
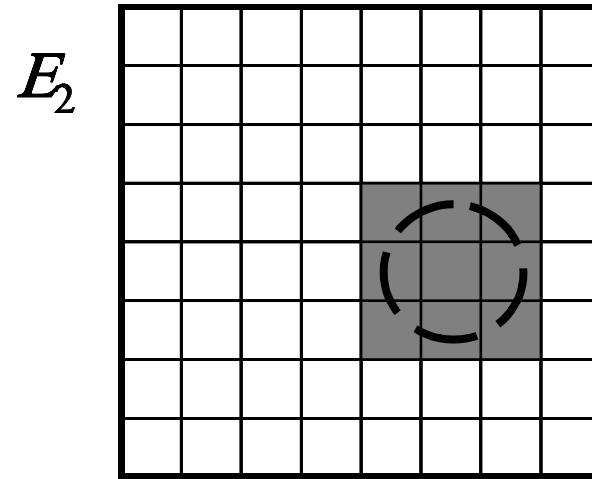
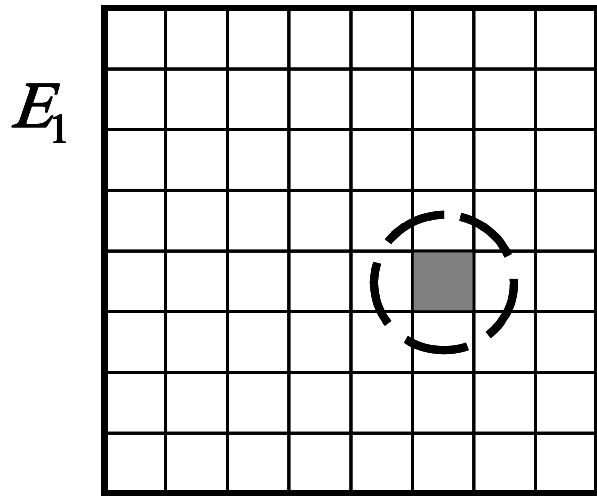
# Complexity of Optimized SLFMM

$$\begin{aligned} F(K_{opt}) &= PN + P^2 \left( \frac{MN}{2P} \right)^{2/3} + PM + PMN \left( \frac{MN}{2P} \right)^{-1/3} \\ &= P(M + N) + (MN)^{2/3} O(P^{4/3}). \end{aligned}$$

At  $K = K_{opt}$ , and  $M = O(N)$ , the complexity of SLFMM is:

$$O(PN + P^{4/3} N^{4/3}) = O(P^{4/3} N^{4/3}).$$

# Hierarchical Spatial Domains



# Multiple Level FMM - 1

- Domain divided hierarchically, starting with  $2^d$  boxes
- Recursively subdivide till  $L$  levels reached
- Upward pass
  - For each source box perform at finest level
    - a multipole expansion of sources in a box, about the box center
    - Expansion valid outside neighborhood of box
  - Combine multipole expansions of all children of a parent to produce multipole expansion for parent
  - Repeat until level 2 reached

# SIS-operator for example

$$S_n(y-x_{*1}) = \sum_{m=0}^{\infty} (S|S)_{mn}(t) S_m(y-x_{*2}), \quad t = x_{*2} - x_{*1}$$

$$\frac{1}{(1-\alpha)^{n+1}} = 1 + (n+1)\alpha + \frac{(n+1)(n+2)}{2!}\alpha^2 + \dots = \sum_{m=0}^{\infty} \frac{(m+n)!}{m!n!}\alpha^m, \quad |\alpha| < 1.$$

$$\begin{aligned} S_n(y-x_{*1}) &= (y-x_{*1})^{-n-1} = (y-x_{*2} - (x_{*1} - x_{*2}))^{-n-1} \\ &= (y-x_{*2})^{-n-1} \left[ 1 - \frac{x_{*1} - x_{*2}}{y-x_{*2}} \right]^{-n-1} \\ &= \sum_{m=0}^{\infty} \frac{(-1)^m (m+n)!}{m!n!} (x_{*2} - x_{*1})^m (y-x_{*2})^{-n-m-1} \\ &= \sum_{m=0}^{\infty} \frac{(-1)^m (m+n)!}{m!n!} t^m S_{n+m}(y-x_{*2}) = \sum_{m=n}^{\infty} \frac{(-1)^{m-n} m!}{n!(m-n)!} t^{m-n} S_m(y-x_{*2}) \end{aligned}$$

$$(S|S)_{mn}(t) = \begin{cases} 0, & m < n \\ \frac{(-1)^{m-n} m!}{n!(m-n)!} t^{m-n}, & m \geq n. \end{cases}$$

# SIS-operator (2)

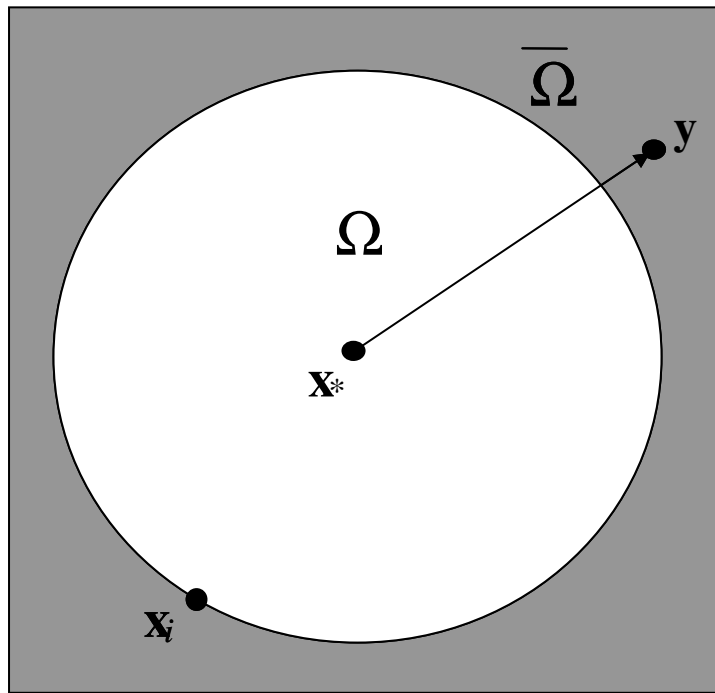
$$S_n(\mathbf{y}-\mathbf{x}_{*1}) = \sum_{m=0}^{\infty} (\mathcal{S}|\mathcal{S})_{mn}(t) S_m(\mathbf{y}-\mathbf{x}_{*2}), \quad t = \mathbf{x}_{*2} - \mathbf{x}_{*1}$$

$$(\mathcal{S}|\mathcal{S})_{mn}(t) = \begin{cases} 0, & m < n \\ \frac{(-1)^{m-n} m!}{n!(m-n)!} t^{m-n}, & m \geq n. \end{cases}$$

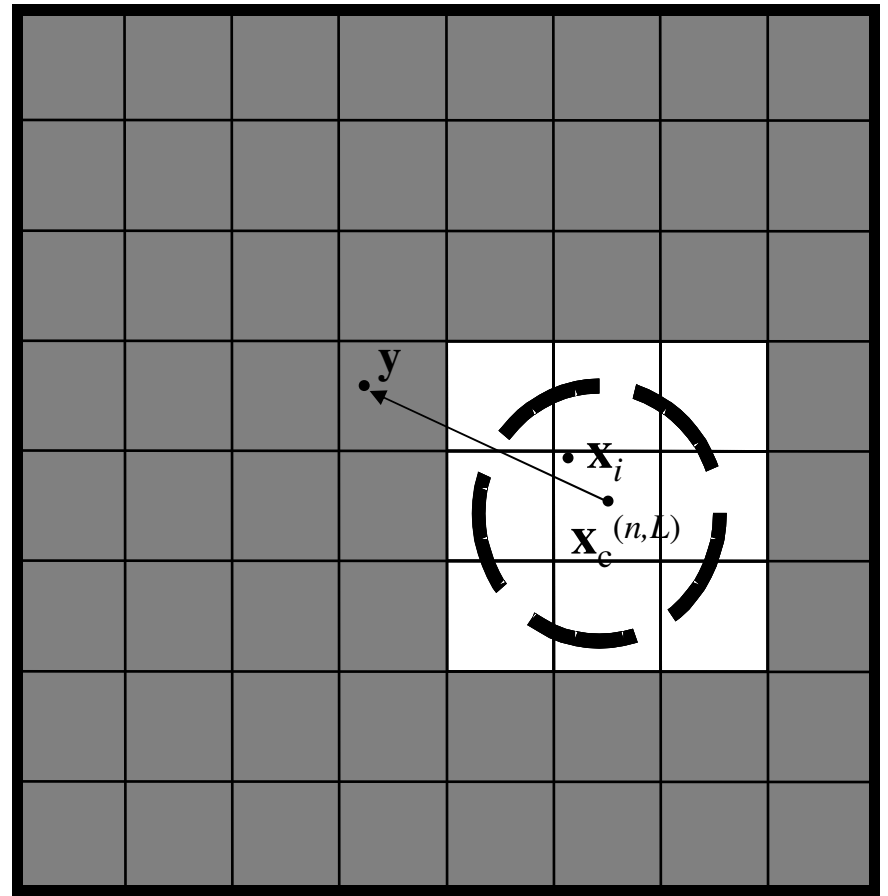
$$(\mathbf{S}|\mathbf{S})(t) = (\mathcal{S}|\mathcal{S})_{mn}(t) = \begin{pmatrix} 1 & 0 & 0 & 0 & \dots \\ -t & 1 & 0 & 0 & \dots \\ t^2 & -2t & 1 & 0 & \dots \\ -t^3 & 3t^2 & -3t & 1 & \dots \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix}$$

# Upward Pass. Step 1.

S-expansion valid in  $\overline{\Omega}$



$E_3$

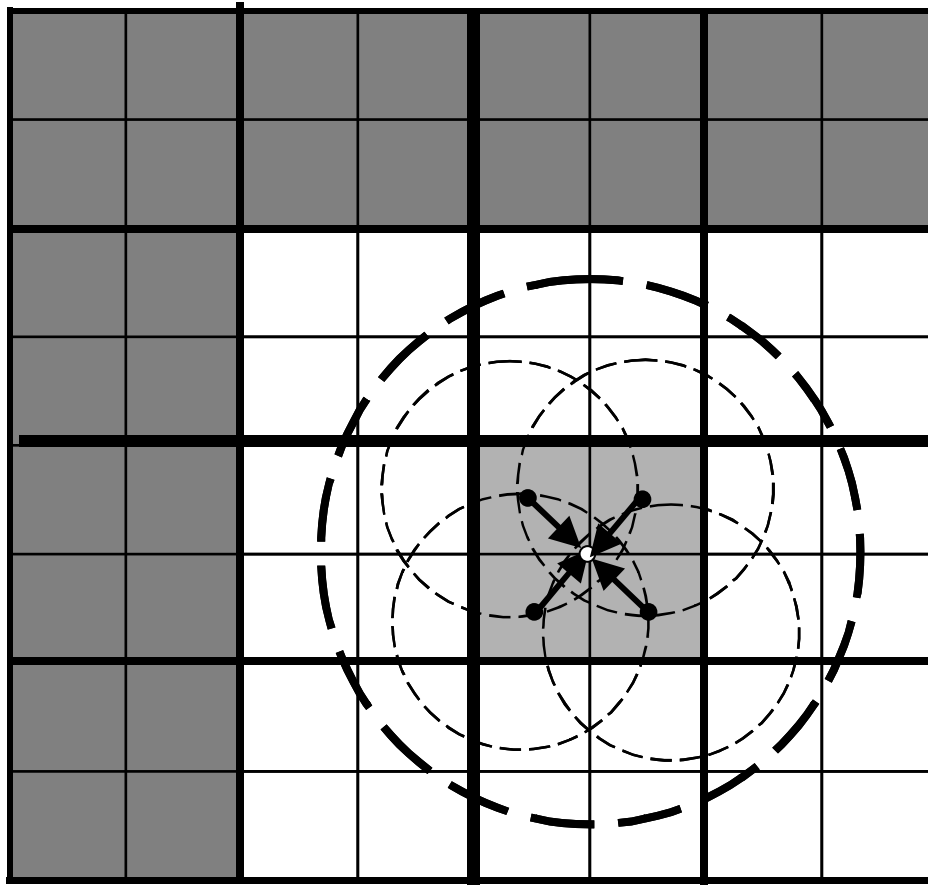


S-expansion valid in  $E_3(n,L)$

# Upward Pass. Step 2.

SIS-translation.

Build potential for the parent box (find its S-expansion).



# Multiple Level FMM - 2

- Downward pass
  - For each receiver box at level 2
  - Initialize coefficients to zero
  - For each non-neighbor source box in the  $E_4$  domain perform a multipole to local SIR translation and add to coefficients
  - R-R translate to children boxes at finer level
  - Add S-R translation of nodes in the new  $E_4$  domain at finer level
  - Repeat until finest level reached
  - Initialize evaluation vector to zero
  - For each evaluation point in the box
    - Evaluate local expansion using coefficients above at each evaluation point and place answer in evaluation vector
    - For each source in the box and its neighborhood evaluate the source contribution directly

# SIR for example case

$$(|y - x_*| < |t|)$$

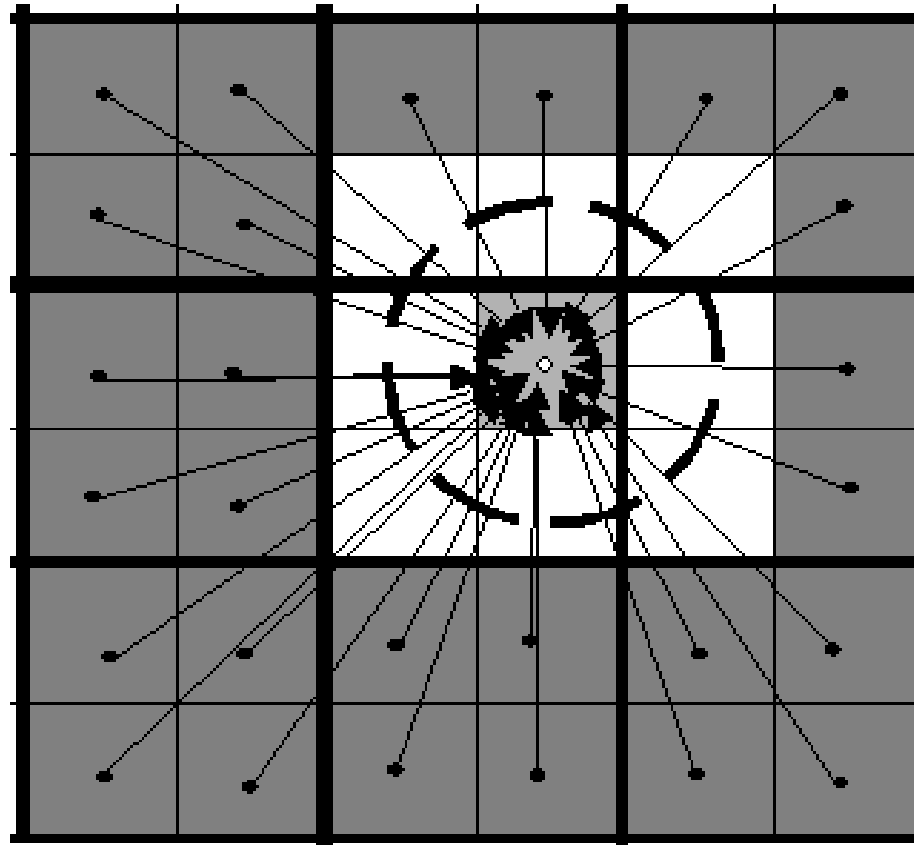
$$\begin{aligned} S_n(y - x_* + t) &= (t + y)^{-n-1} = \sum_{m=0}^{\infty} \frac{1}{m!} \frac{d^m S_n(t)}{dt^m} (y - x_*)^m \\ &= \sum_{m=0}^{\infty} \frac{1}{m!} \frac{d^m S_n(t)}{dt^m} R_m(y - x_*) = \sum_{m=0}^{\infty} (S|R)_{mn}(t) R_m(y - x_*). \end{aligned}$$

So

$$(S|R)_{mn}(t) = \frac{1}{m!} \frac{d^m S_n(t)}{dt^m} = \frac{(-1)^m (m+n)!}{m! n! t^{n+m+1}}.$$

$$(S|R)(t) = \begin{pmatrix} t^{-1} & t^{-2} & t^{-3} & \dots \\ -t^{-2} & -2t^{-3} & -3t^{-4} & \dots \\ t^{-3} & 3t^{-4} & 6t^{-5} & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix}$$

# SIR-translation



# R|R-operator for example case

$$R_n(y-x_{*1}) = \sum_{m=0}^{\infty} (R|R)_{mn}(t) R_m(y-x_{*2}), \quad t = x_{*2} - x_{*1}$$

$$\begin{aligned} R_n(y-x_{*1}) &= (y-x_{*1})^n = (y-x_{*2} - (x_{*1}-x_{*2}))^n \\ &= \sum_{m=0}^n \frac{(-1)^m n!}{m!(n-m)!} (x_{*2}-x_{*1})^{n-m} (y-x_{*2})^m \\ &= \sum_{m=0}^n \frac{(-1)^m n!}{m!(n-m)!} t^{n-m} R_m(y-x_{*2}). \end{aligned}$$

$$(R|R)_{mn}(t) = \begin{cases} 0, & m > n \\ \frac{(-1)^m n!}{m!(n-m)!} t^{n-m}, & m \leq n. \end{cases}$$

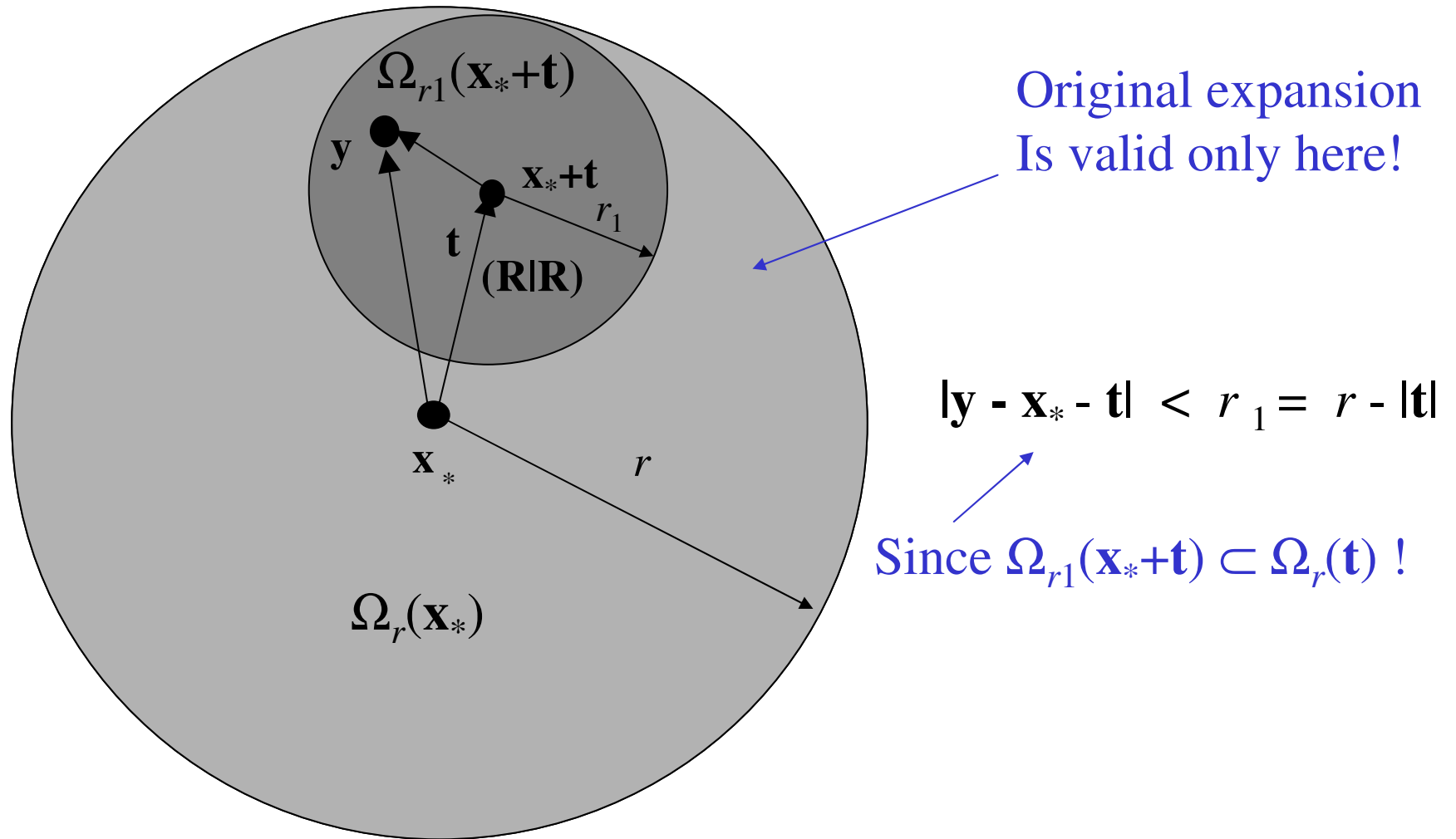
# R|R-operator(2)

$$R_n(y-x_{*1}) = \sum_{m=0}^n (R|R)_{mn}(t) R_m(y-x_{*2}), \quad t = x_{*2} - x_{*1}$$

$$(R|R)_{mn}(t) = \begin{cases} 0, & m > n \\ \frac{(-1)^m n!}{m!(n-m)!} t^{n-m}, & m \leq n. \end{cases}$$

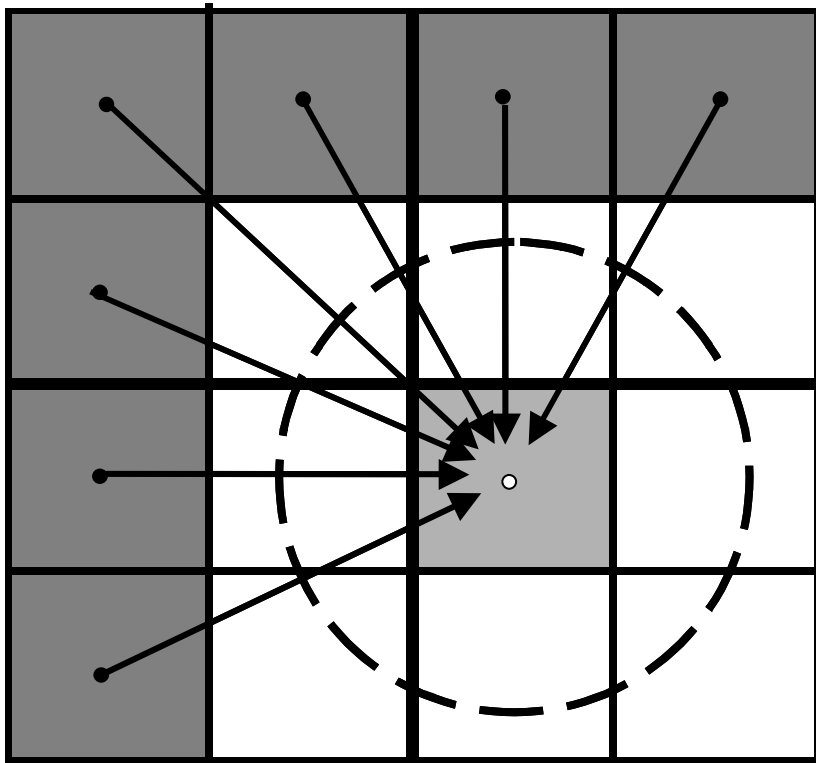
$$(\mathbf{R}|\mathbf{R})(t) = (R|R)_{mn}(t) = \begin{pmatrix} 1 & t & t^2 & t^3 & \dots \\ 0 & -1 & -2t & -3t^2 & \dots \\ 0 & 0 & 1 & 3t & \dots \\ 0 & 0 & 0 & -1 & \dots \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix}$$

# RIR-reexpansion of the same function over shifted basis (2)

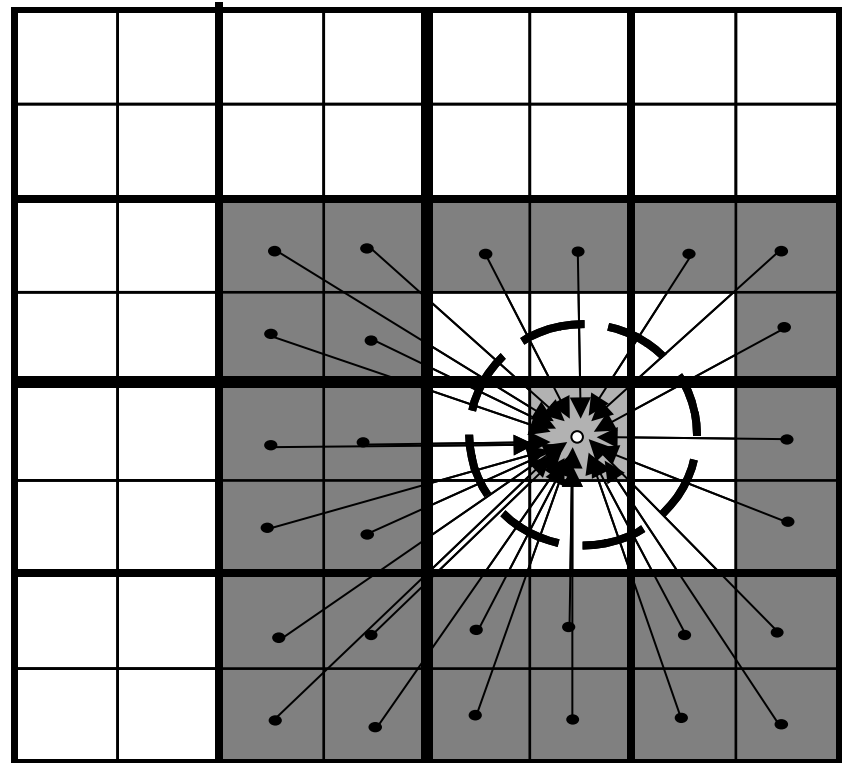


# Downward Pass. Step 1.

Level 2:

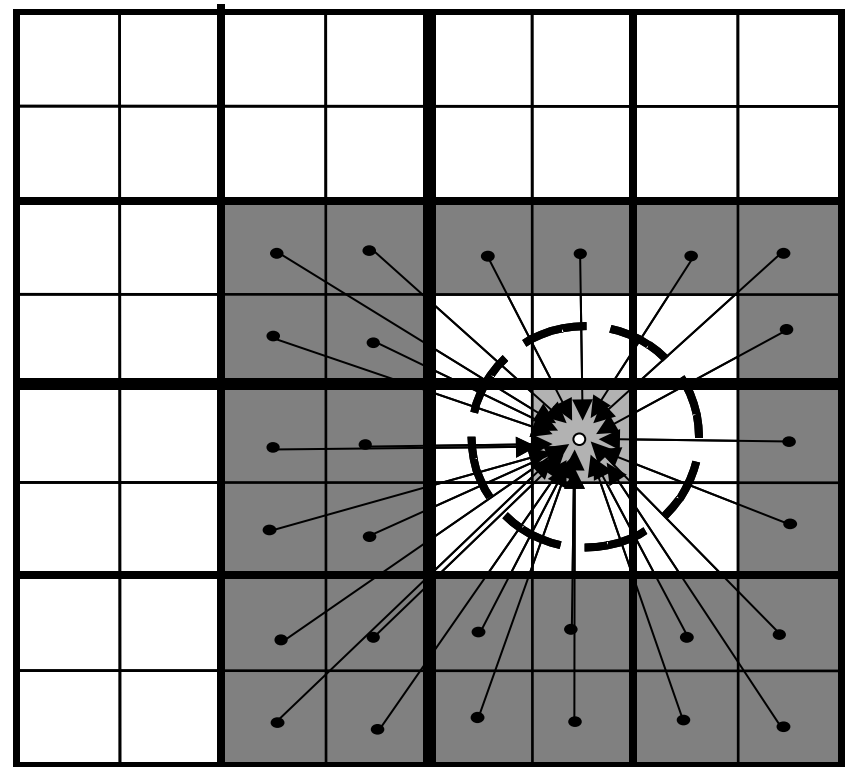


Level 3:



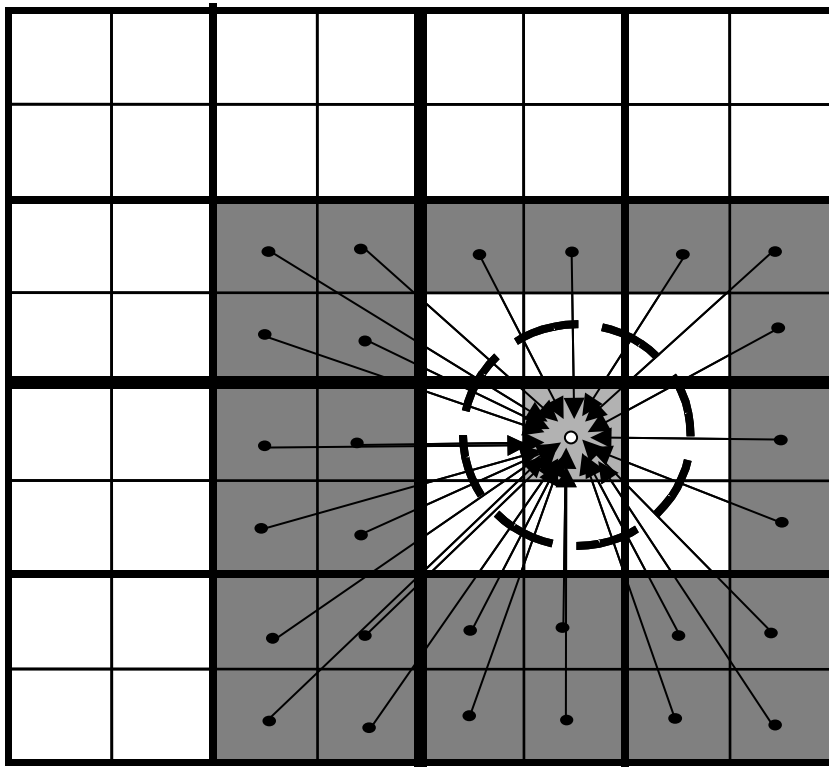
# Downward Pass. Step 1.

THIS IS USUALLY THE  
MOST EXPENSIVE STEP OF  
THE ALGORITHM



# Downward Pass. Step 1.

$$P_4 = \text{PowerOfE}_4\text{Neighborhood} = 3^d 2^d - 3^d = 3^d (2^d - 1)$$



$$d = 1 : P_4 = 3,$$

$$d = 2 : P_4 = 27,$$

$$d = 3 : P_4 = 189$$

Exponential  
Growth

Total number of SIR-translations  
per 1 box in  $d$ -dimensional space

(not on the domain boundary)

It is worth to think about optimizations

# Downward Pass. Step 2.

**Step 2.** At  $l = 2$  we have

$$\Phi_3^{(n,2)}(\mathbf{y}) = \Phi_4^{(n,2)}(\mathbf{y}), \quad \mathbf{D}^{(n,2)} = \tilde{\mathbf{D}}^{(n,2)},$$

Form  $\Phi_3^{(n,l)}(\mathbf{y})$  (or expansion coefficients of this function) by adding  $\Phi_4^{(Parent(n),l-1)}(\mathbf{y})$  to  $(\mathbf{R}|\mathbf{R})$ -translated coefficients of the parent box to the child center:

$$\Phi_3^{(n,l)}(\mathbf{y}) = \mathbf{D}^{(n,l)} \circ \mathbf{R}(\mathbf{y} - \mathbf{x}_c^{(n,l)}),$$

$$\mathbf{D}^{(n,l)} = \tilde{\mathbf{D}}^{(n,l)} + (\mathbf{R}|\mathbf{R}) \left( \mathbf{x}_c^{(n,l)} - \mathbf{x}_c^{(m,l-1)} \right) \mathbf{D}^{(m,l-1)}, \quad m = Parent(n).$$

# Downward Pass. Step 2.

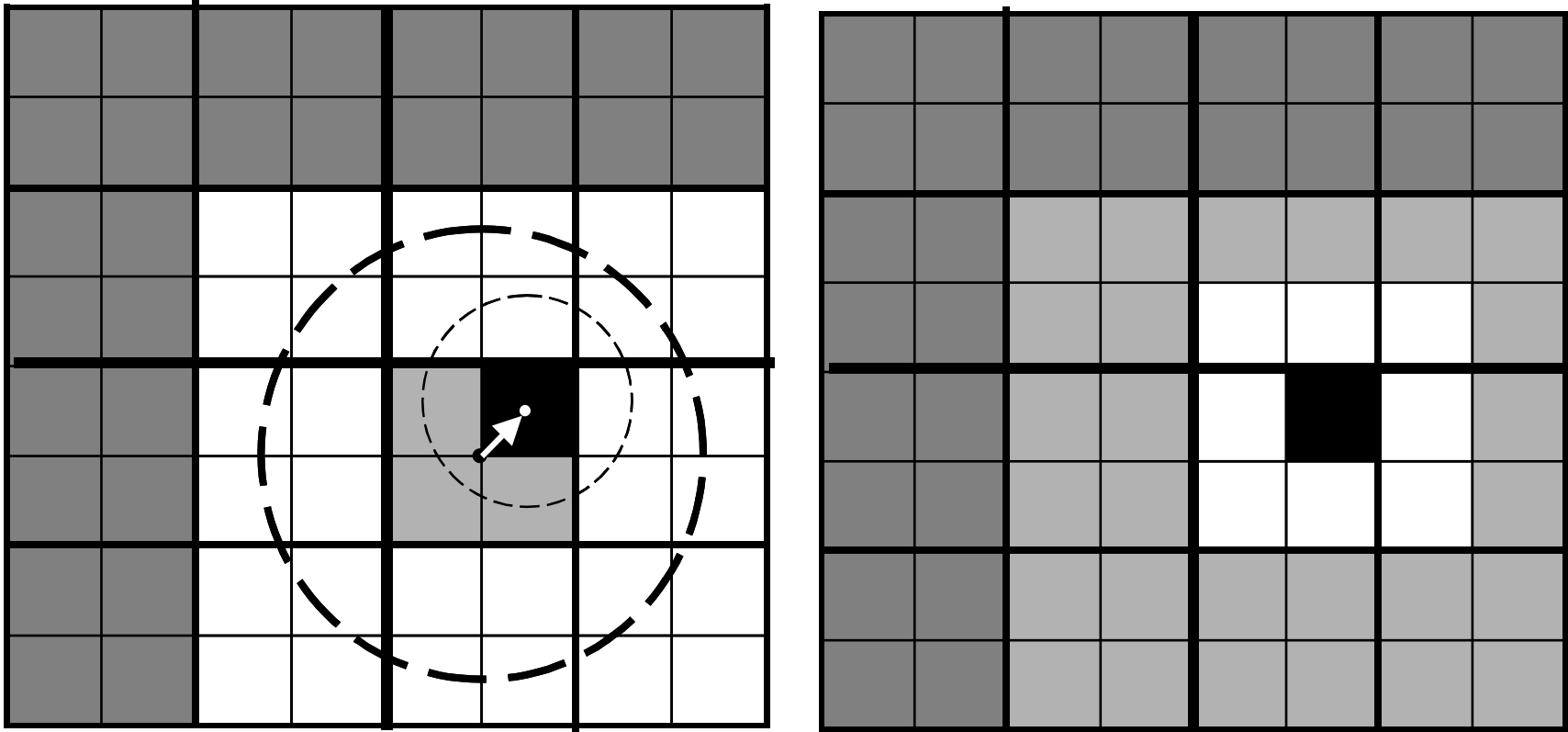


Figure shows that local-to-local translation is applicable in this case (smaller sphere is located completely inside the larger sphere), and junction of structures  $E_3(n, l)$  and  $E_4(n, l + 1)$  produces  $E_3(n, l + 1)$  :

$$E_3(n, l + 1) = E_3(n, l) \cup E_4(n, l + 1).$$

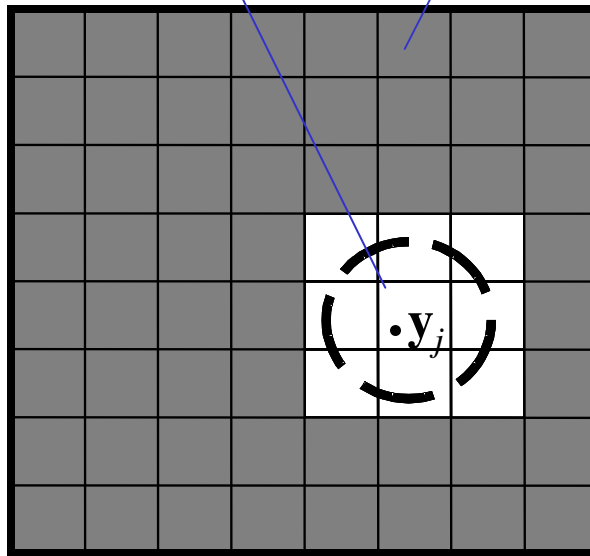
# Final Summation

As soon as coefficients  $\mathbf{D}^{(n,L)}$  are determined total potential can be computed for any point  $\mathbf{y}_j \in E_1(0,0)$ , where  $\Phi_2^{(n,l)}(\mathbf{y})$  can be computed straightforward. So:

$$v_j = \Phi(\mathbf{y}_j) = \sum_{\mathbf{x}_i \in E_2(n,L)} u_i \Phi(\mathbf{y}_j, \mathbf{x}_i) + \mathbf{D}^{(n,L)} \circ \mathbf{R}(\mathbf{y}_j - \mathbf{x}_c^{(n,L)}), \quad \mathbf{y}_j \in E_1(n,L).$$

Contribution of  $E_2$

Contribution of  $E_3$



# Itemized Cost of MLFMM

Regular mesh:

$$N = 2^{L_*d}, \quad s = 2^{L_s d}, \quad L = L_{\max} = L_* - L_s.$$

Assume that all translation costs are the same,  
 $CostTranslation(P)$

$$CostUpward_1 = N CostExpansion(P) = O(NP).$$

$$CostUpward_2 = 2^d (2^{(L-1)d} + 2^{(L-2)d} + \dots + 2^{2d}) CostSS(P)$$

$$< \frac{2^d}{2^d - 1} (2^{Ld} - 1) CostSS(P) \sim \frac{N}{s} CostSS(P)$$

$$CostDownward_1 \lesssim P_4(d) (2^{2d} + \dots + 2^{Ld}) CostSR(P) \sim P_4(d) \frac{N}{s} CostSR(P),$$

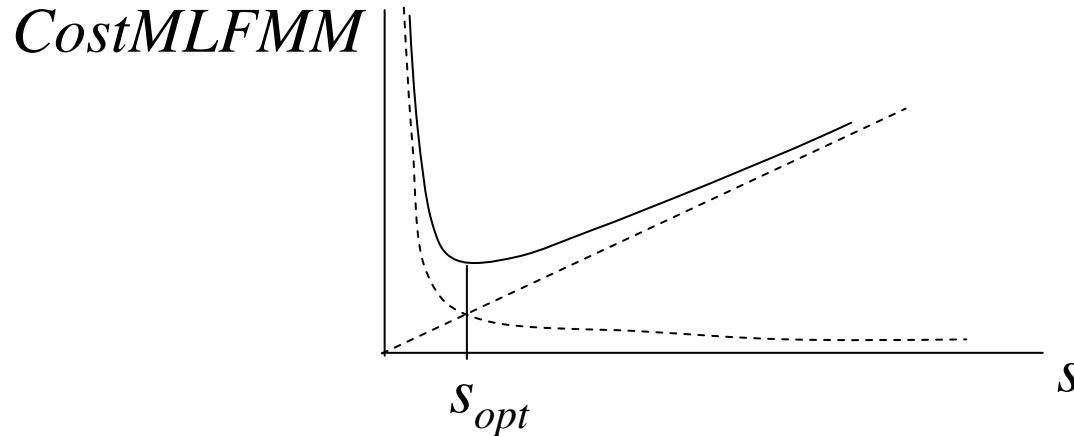
$$CostDownward_2 = 2^d (2^{2d} + \dots + 2^{(L-1)d}) CostRR(P) \sim \frac{N}{s} CostRR(P),$$

$$CostEvaluation = M(P_2(d)s CostFunc + P).$$

Powers of  $E_4$   
 and  $E_2$  neighborhoods

$$CostMLFMM = (M + N)P + (P_4(d) + 2) \frac{N}{s} CostTranslation(P) + P_2(d)sM CostFunc$$

# Optimization of the Grouping Parameter



$$CostMLFMM = (M + N)P + (P_4(d) + 2) \frac{N}{s} CostTranslation(P) + P_2(d) s M CostFunc$$

$$\frac{\partial CostMLFMM}{\partial s} = -(P_4(d) + 2) \frac{N}{s^2} CostTranslation(P) + P_2(d) M CostFunc = 0$$

$$s_{opt} = \left[ \frac{N(P_4(d) + 2) CostTranslation(P)}{M P_2(d) CostFunc} \right]^{1/2}$$

$$CostMLFMM_{opt} = (M + N)P + 2 [MN(P_4(d) + 2) P_2(d) CostTranslation(P) CostFunc]^{1/2}$$

# Optimization of the Grouping Parameter (Example)

$$s_{opt} = \left[ \frac{N(P_4(d) + 2)CostTranslation(P)}{MP_2(d)CostFunc} \right]^{1/2}.$$

$$CostMLFMM_{opt} = (M + N)P + 2[MN(P_4(d) + 2)P_2(d)CostTranslation(P)CostFunc]^{1/2}.$$

Example:

$$N = M, \quad P_4(d) = 3^d(2^d - 1), \quad P_2(d) = 3^d,$$

$$CostTranslation(P) = P^2, \quad CostFunc = 1$$

$$s_{opt} \sim 2^{d/2}P, \quad CostMLFMM_{opt} \sim 2NP(1 + 3^d 2^{d/2})$$

$$For d = 2, \quad P = 10, \quad s_{opt} \sim 38, \quad CostMLFMM_{opt} \sim 38NP = 380N.$$

If non-optimized,

$$s = 1; \quad CostMLFMM_{opt} \sim NP(2 + 3^d 2^d P)$$

$$For d = 2, \quad P = 10, \quad s = 1, \quad CostMLFMM_{opt} \sim 360NP = 3600N.$$

In this example optimization results in about 10 times savings!

# Domains of Expansion Validity (1)

Consider  $d$ -dimensional space.

Size of box at level  $l$  :

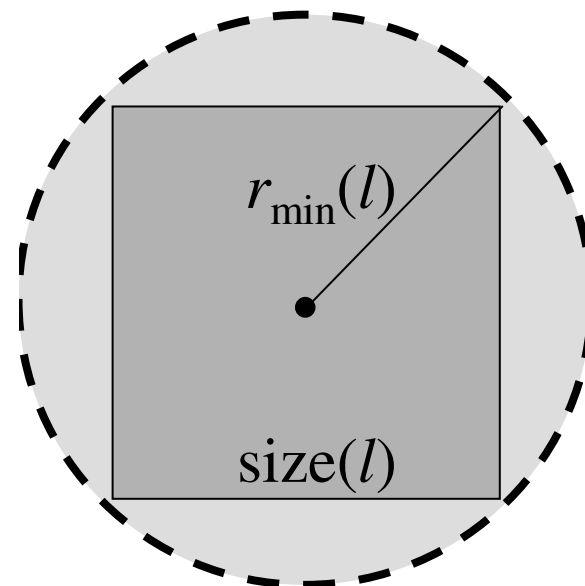
$$size(l) = 2^{-l}.$$

Half of diagonal of box at level  $l$  :

$$diag(l) = 2^{-l} \frac{1}{2} \sqrt[d]{d} = 2^{-l-1} \sqrt[d]{d}.$$

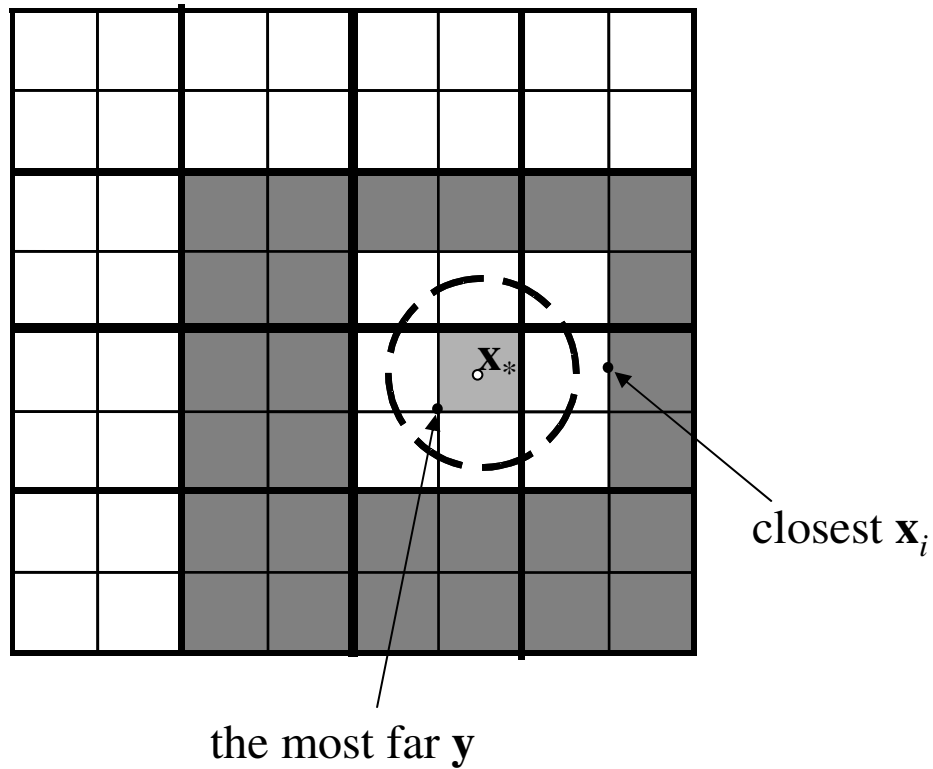
Minimum sphere radius containing the box

$$r_{\min}(l) = diag(l).$$

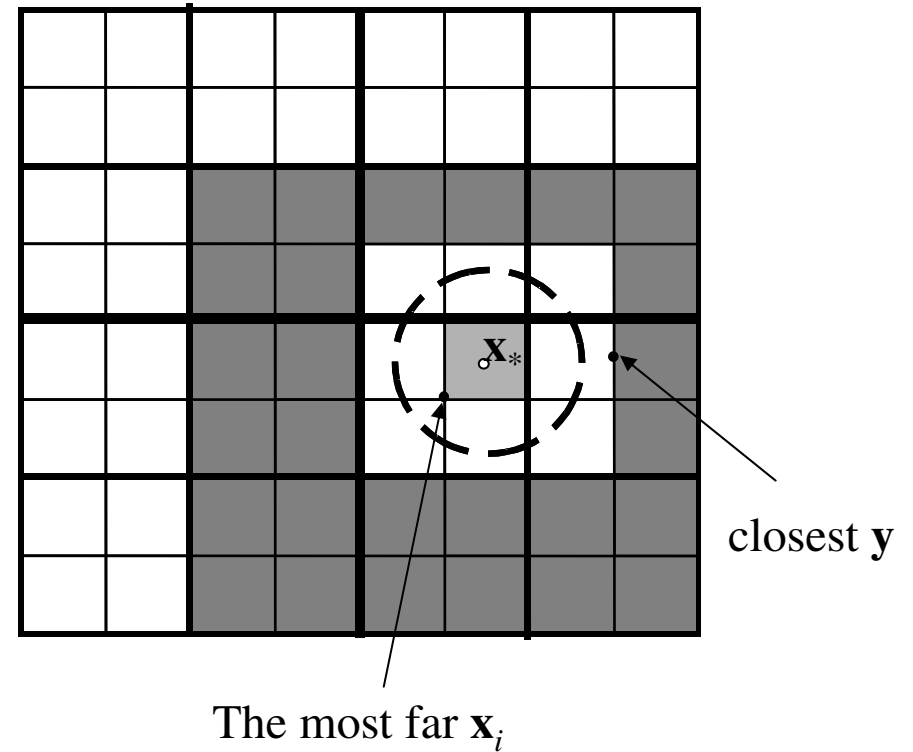


# Domains of Expansion Validity (2)

R-expansion



S-expansion



# Domains of Expansion Validity (3).

## R-expansion.

- Potentials (functions) can be factorized as (local expansion):

$$\Phi(\mathbf{x}_i, \mathbf{y}) = \mathbf{A}(\mathbf{x}_i, \mathbf{x}_*) \circ \mathbf{R}(\mathbf{y} - \mathbf{x}_*), \quad |\mathbf{y} - \mathbf{x}_*| < r < |\mathbf{x}_i - \mathbf{x}_*|, \quad i = 1, \dots, N$$

$$|\mathbf{y} - \mathbf{x}_*| \leq r_{\min}(l),$$

$$|\mathbf{x}_i - \mathbf{x}_*| \geq \frac{3}{2} \text{size}(l).$$

Strict inequality:

$$r_{\min}(l) < \frac{3}{2} \text{size}(l)$$

$$2^{-l-1} \sqrt{d} < \frac{3}{2} 2^{-l}, \quad d < 9.$$

# Domains of Expansion Validity (8). R|R and S|S-translations.

- $R$ -expansion coefficients can be  $R|R$ -translated:

$$|y - x_{*2}| < |x_i - x_{*1}| - |x_{*1} - x_{*2}| :$$

$$A(x_i, x_{*2}) = (R|R)(x_{*2} - x_{*1})A(x_i, x_{*1})$$

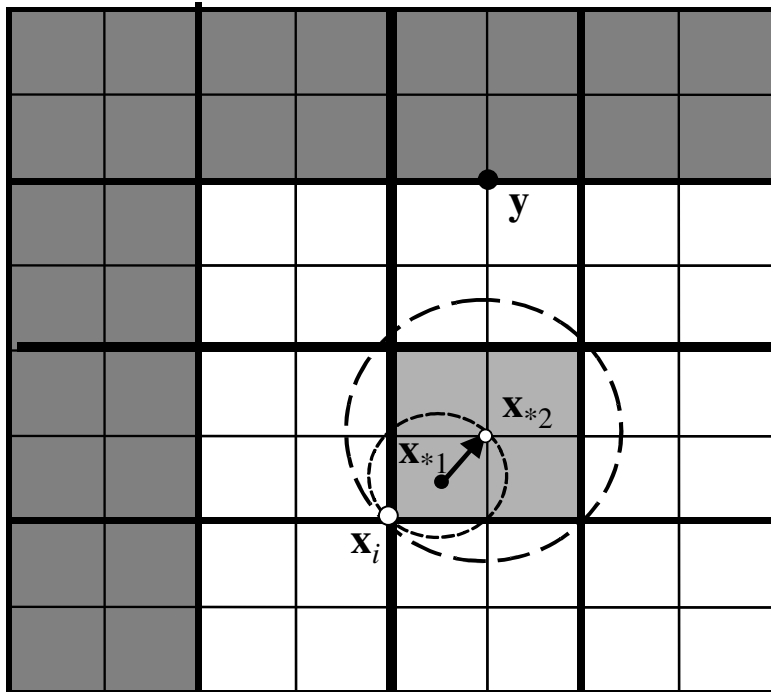
- $S$ -expansion coefficients can be  $S|S$ -translated:

$$|y - x_{*2}| > |x_{*1} - x_{*2}| + |x_i - x_{*1}|,$$

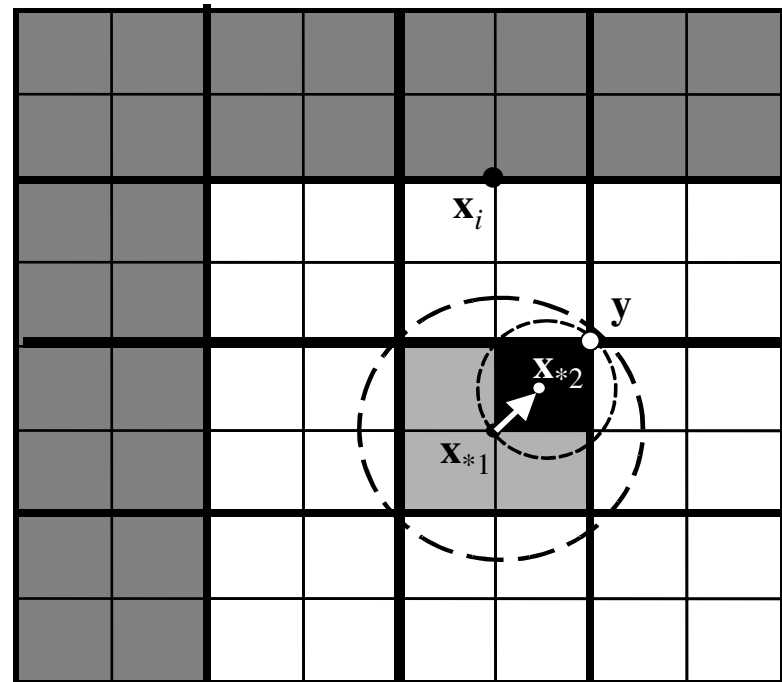
$$B(x_i, x_{*2}) = (S|S)(x_{*2} - x_{*1})B(x_i, x_{*1})$$

No  
additional  
constraints

SIS



R|R

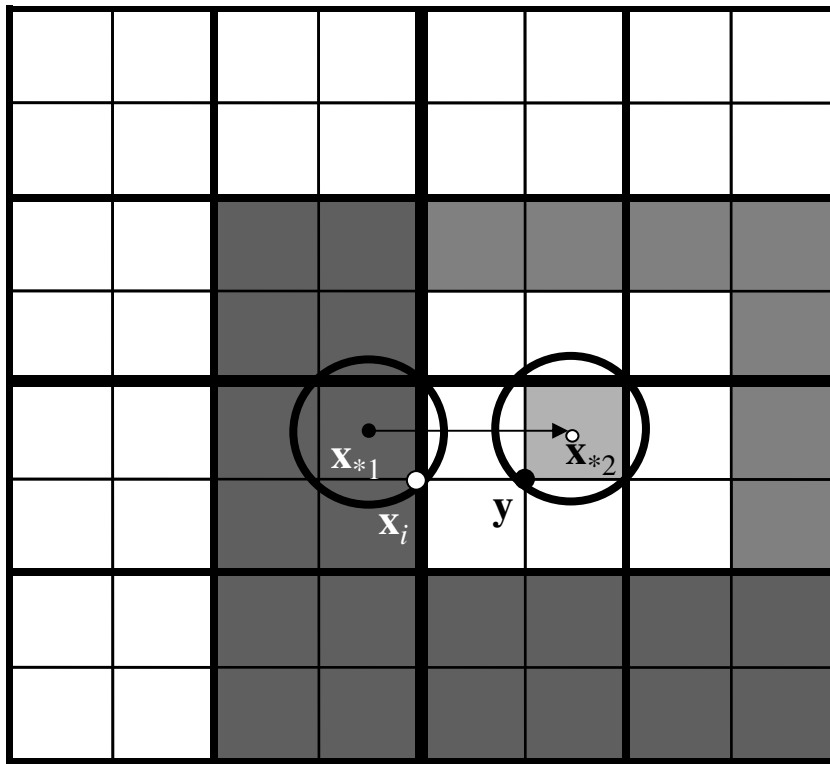


# Domains of Expansion Validity (9). S|R-translation.

- $S$ -expansion coefficients can be  $S|R$ -translated (converted to  $R$ -expansion coefficients)

$$|\mathbf{y} - \mathbf{x}_{*2}| < |\mathbf{x}_{*1} - \mathbf{x}_{*2}| - |\mathbf{x}_i - \mathbf{x}_{*1}|,$$

$$\mathbf{A}(\mathbf{x}_i, \mathbf{x}_{*2}) = (\mathbf{S|R})(\mathbf{x}_{*2} - \mathbf{x}_{*1})\mathbf{B}(\mathbf{x}_i, \mathbf{x}_{*1})$$



$$|\mathbf{y} - \mathbf{x}_*| \leq r_{\min}(l), \quad |\mathbf{x}_i - \mathbf{x}_*| \leq r_{\min}(l),$$

$$\min|\mathbf{x}_{*1} - \mathbf{x}_{*2}| = 2\text{size}(l).$$

$$2^{-l-1}\sqrt{d} + 2^{-l-1}\sqrt{d} \leq 2^{-l+1}, \quad d \leq 4.$$

$d < 4$  – strict,

$d \leq 4$  – weak.