CMSC 858M/AMSC 698R

Fast Multipole Methods

Nail A. Gumerov & Ramani Duraiswami

Lecture 6

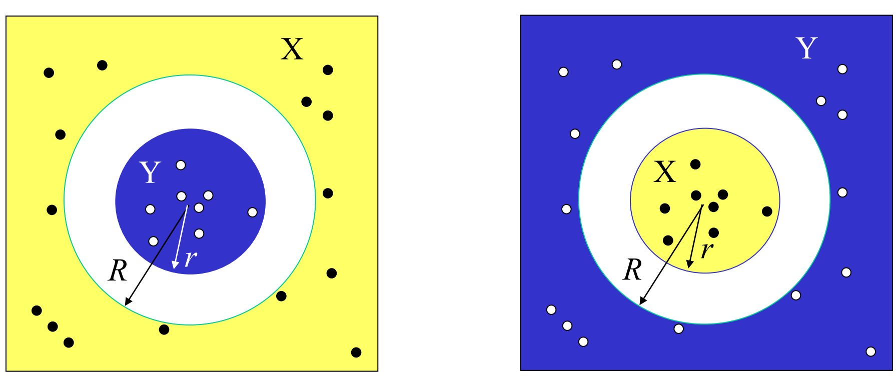
Outline

- Spatial Grouping: One of key stones of the FMM
- Natural spatial grouping. Well separated sets.
- Problem of "outliers". Modifications of "Middleman".
- ``Pre-FMM"- universal fast algorithm
- Space partitioning with respect to the target Set
- Optimization of the ``Pre-FMM"
- Space partitioning with respect to the source set

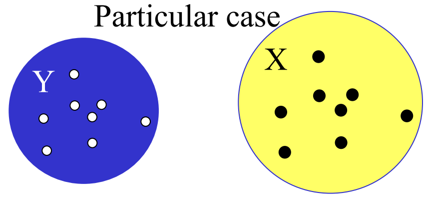
Well separated sets

Definition: Two sets of points in R^d, X and Y, are called well separated, if there exist two co-centric spheres of radii r and R, $r < R$, such that all points of Y are located inside the smaller sphere, and there are no points of X located inside the larger sphere. (In this definition sets X and Y can be exchanged).

Well separated sets (examples)



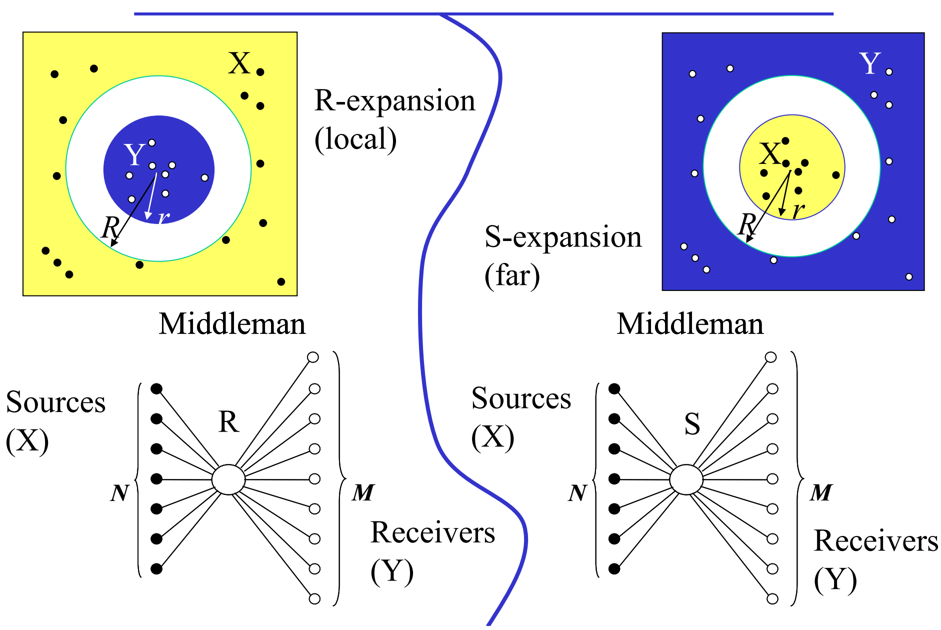
Particular case



Can we prove that…

- For singular factorizable kernel and well separated sets of the sources and targets, the matrix-vector multiplication can be performed using the ``Middleman'' algorithm?

Middleman for well separated sets



Pecularities of ``Middleman” for singular kernels

- Separation of sets is crucial;
- Type of factorization (S or R) depends on the type of source/receiver distribution;
- Separation parameter, *r/R* controls the convergence of the series and for given accuracy the truncation number substantially depends on this parameter (so the efficiency of the fast summation method).

Example of the error bound

$$\Phi(y,x) = \frac{1}{y-x}, \quad |y-x_*| < r, \quad |x-x_*| > R.$$

We have

$$\Phi(y,x) = -\frac{1}{x-x_*}\sum_{n=0}^{\infty}\frac{(y-x_*)^n}{(x-x_*)^n} = -\frac{1}{x-x_*}\sum_{n=0}^{p-1}\frac{(y-x_*)^n}{(x-x_*)^n} + \epsilon_p.$$

The residual can be computed exactly:

$$\epsilon_p = -\frac{1}{x-x_*}\sum_{n=p}^{\infty}\frac{(y-x_*)^n}{(x-x_*)^n} = \frac{(y-x_*)^p}{(x-x_*)^p}\left[-\frac{1}{x-x_*}\sum_{n=p}^{\infty}\frac{(y-x_*)^{n-p}}{(x-x_*)^{n-p}}\right]$$

$$= \frac{(y-x_*)^p}{(x-x_*)^p}\left[-\frac{1}{x-x_*}\sum_{n=0}^{\infty}\frac{(y-x_*)^n}{(x-x_*)^n}\right] = \frac{(y-x_*)^p}{(x-x_*)^p}\Phi(y,x).$$

$$|\Phi(y,x) - \Phi^{(p)}(y,x)| \leqslant |\epsilon_p| = \frac{|y-x_*|^p}{|x-x_*|^p}|\Phi(y,x)| \leqslant \left(\frac{r}{R}\right)^p |\Phi(y,x)|.$$

Relative error is bounded by $(r/R)^p$ and absolute error is bounded by

$$|\epsilon_p| \leqslant \left(\frac{r}{R}\right)^p \max\frac{1}{|y-x|} \leqslant \frac{1}{R-r}\left(\frac{r}{R}\right)^p.$$

Model of geometric error bound for higher dimensionalities

Single source error:

$$|\epsilon_p| \leqslant A\left(\frac{r}{R}\right)^p$$

Error for sum of N-sources (assume $\max_i |u_i| = 1$)

$$|\epsilon| \leqslant \left|\sum_{i=1}^{N} u_i \epsilon_p\right| \leqslant \sum_{i=1}^{N} |u_i||\epsilon_p| = |\epsilon_p| \sum_{i=1}^{N} |u_i| \leqslant N|\epsilon_p| \max_i |u_i| \leqslant NA\left(\frac{r}{R}\right)^p.$$

Then

$$p \geqslant \frac{\log \frac{NA}{|\epsilon|}}{\log\left(\frac{R}{r}\right)}.$$

If $\max_i |u_i| = 1/N$:

$$p \geqslant \frac{\log \frac{A}{|\epsilon|}}{\log\left(\frac{R}{r}\right)}.$$

Actual complexity of "Middleman"

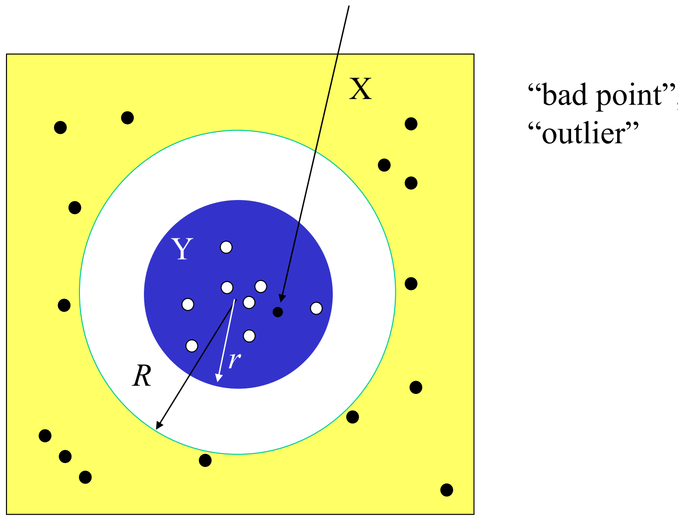
Assume $M \sim N$ and $p \sim \log N + \log \frac{1}{\epsilon}$. Then complexity of the "Middleman" is

$$C = O(pN) = O(N \log N + N \log \frac{1}{\epsilon}).$$

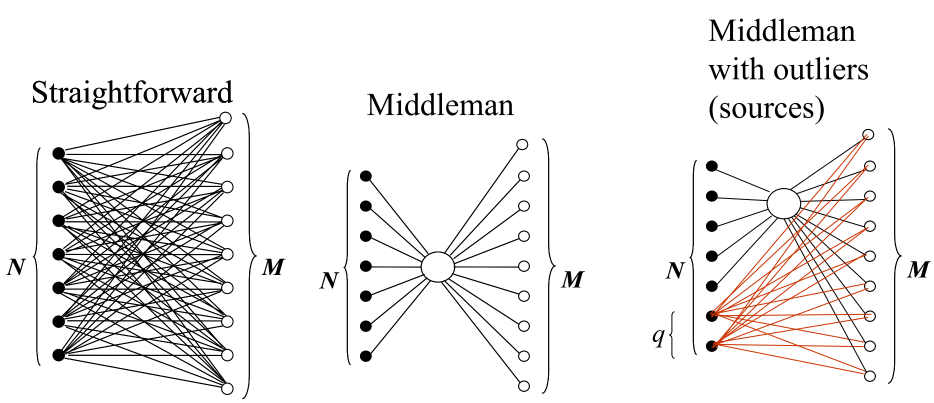
For $p \sim \log \frac{1}{\epsilon}$ we have

$$C = O(pN) = O(N \log \frac{1}{\epsilon}).$$

One point that spoils algorithm…



Modification of the “Middleman” for outliers

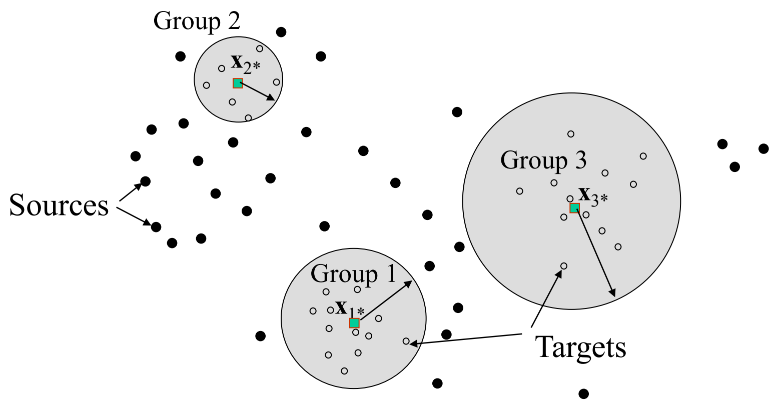


Complexity: $O(NM)$

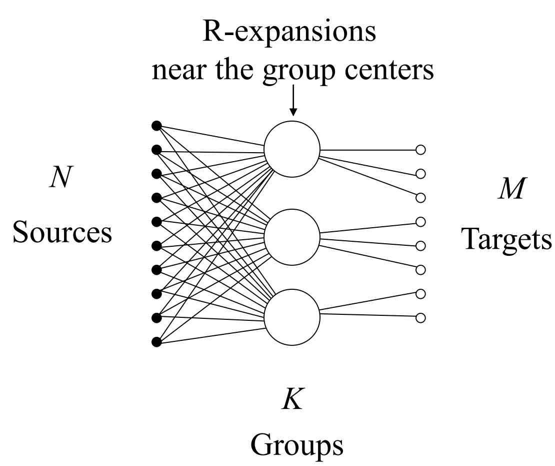
Complexity: $O(pN+pM)$

Complexity: $O(p(N\text{-}q)+pM+qM)$

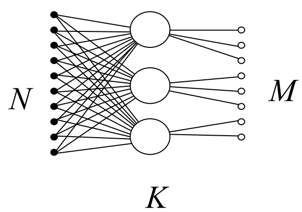
Natural spatial grouping (grouping with respect to the target set)



Natural spatial grouping (continuation)



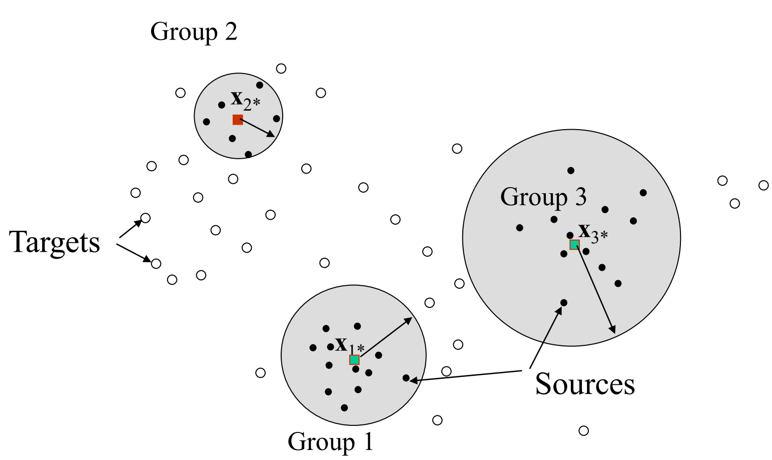
Natural spatial grouping (continuation)



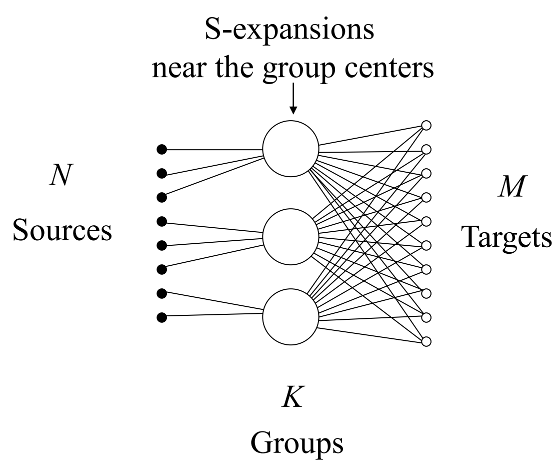
Asymptotic Complexity:

1) Let the R-expansion has p-terms;
2) To build them for K groups we need $O(pNK)$ operations.
3) To evaluate them we need $O(pM)$ operations.
4) Total complexity: $O(p(NK+M))$.
5) Better then the Straightforward method, if $pK \ll M$. In this case $p(NK+M) \ll NM$

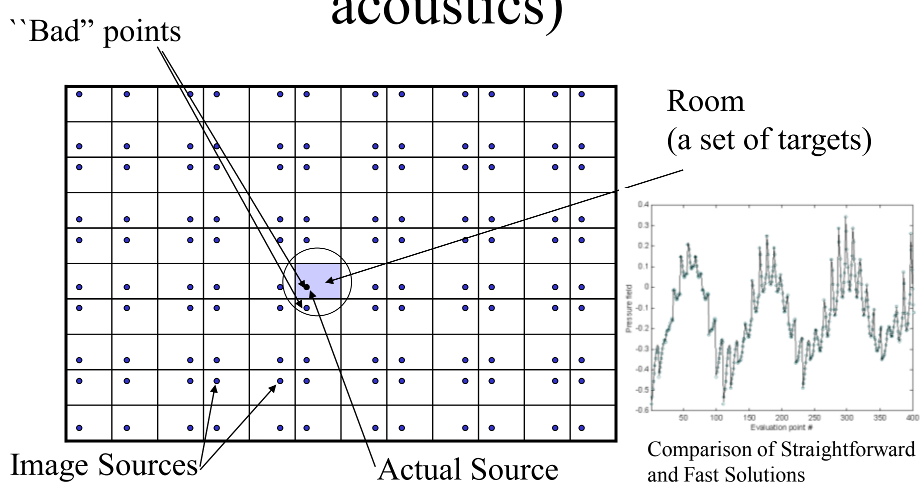
Natural spatial grouping for (Grouping with respect to the source set)



Natural spatial grouping (continuation)



Outliers (an example from room acoustics)



Comparison of Straightforward and Fast Solutions

(R. Duraiswami, N.A. Gumerov, D.N. Zotkin & L.S. Davis, Efficient Evaluation Of Reverberant Sound Fields, 2001 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2001).

Outliers (continued)

Universal Recipe: If the number of the outliers is small, then compute their contribution directly.

E.g. if this number is smaller than p,
then the outliers do not change the algorithm complexity.

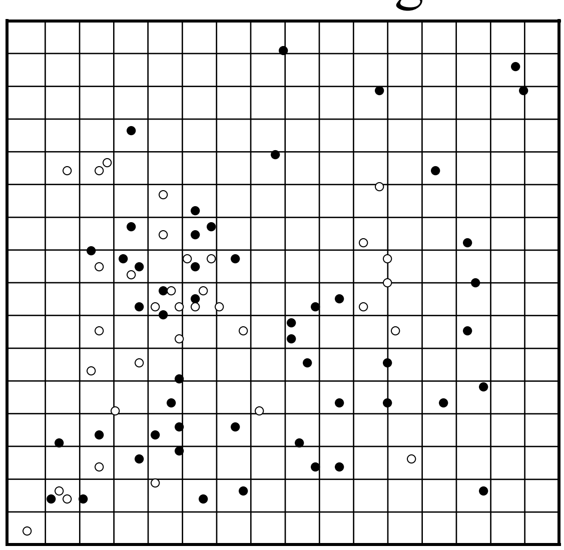
Examples of natural spatial grouping

- Stars (form galaxies, gravity);
- Flow past a body (vortices are grouped in a wake);
- Statistics (clusters of statistical data points);
- People (Organized in groups, cities, etc.);
- Create your own example !

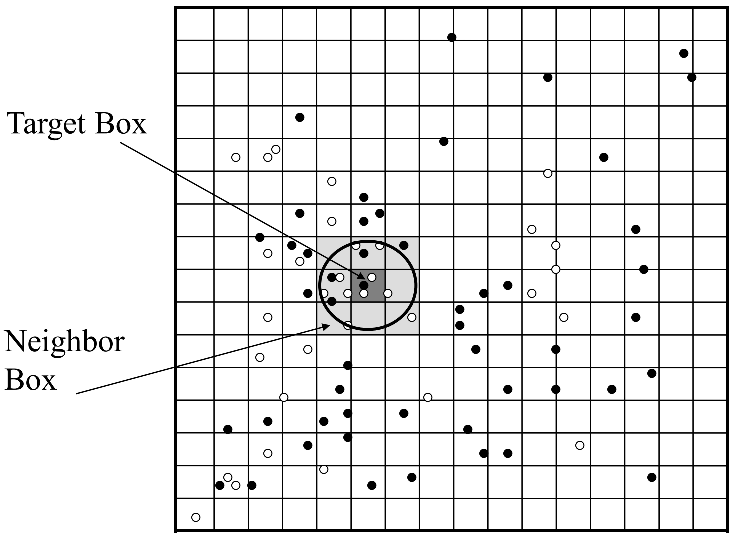
Deficiencies

- Data points may be not naturally grouped;
- Need intelligence to identify the groups: Problem with the algorithms (Artificial Intelligence?)
- Problem dependent.

The Answer is: Space Partitioning



Space partitioning with respect to the target set



An algorithm for computation with space partitioning (Pre-FMM)

- Decomposition of the sum:

Singular Part (sources in the neighborhood)

$$v(\mathbf{y}_j) = \sum_{\mathbf{x}_i \in R_n^+} u_i \Phi(\mathbf{y}_j - \mathbf{x}_i) + \sum_{\mathbf{x}_i \in R_n^-} u_i \Phi(\mathbf{y}_j - \mathbf{x}_i), \quad \mathbf{y}_j \in R_n.$$

Regular Part (sources outside the neighborhood)

- Factorization of the regular part

$$\Phi(\mathbf{y}_j - \mathbf{x}_i) = \sum_{m=0}^{p-1} a_m(\mathbf{x}_i, \mathbf{x}_{n*}) R_m(\mathbf{y}_j - \mathbf{x}_{n*}) + Error_p, \quad \mathbf{y}_j, \mathbf{x}_{n*} \in R_n, \quad \mathbf{x}_i \in R_n^-.$$

- Fast computation of the regular part

$$\sum_{\mathbf{x}_i \in R_n^-} u_i \Phi(\mathbf{y}_j - \mathbf{x}_i) = \sum_{m=0}^{p-1} \left[\sum_{\mathbf{x}_i \in R_n^-} u_i a_m(\mathbf{x}_i, \mathbf{x}_{n*}) \right] R_m(\mathbf{y}_j - \mathbf{x}_{n*}).$$

- Direct summation of the singular part, $\sum_{\mathbf{x}_i \in R_n^+} u_i \Phi(\mathbf{y}_j - \mathbf{x}_i)$

Asymptotic complexity of the Pre-FMM

- Let N be the number of sources, M the number of targets, and K the number of target boxes.
- Each target box, R_n, M_n targets, $n = 1,\dots,K$.
- The *neighborhood* of each target box contains N_n sources, $n = 1,\dots,K$.
- Computation of the expansion coefficients for the regular part for the nth box requires $O((N-N_n)p)$ operations.
- Evaluation of the regular expansion for the nth box requires $O(M_n p)$ operations.
- Direct computation of the singular part requires $O(M_n N_n)$ operations.
- Total complexity is:

$$Complexity = O\left(\sum_{n=1}^{K}[(N-N_n)p + M_n p + M_n N_n]\right).$$

Asymptotic Complexity of the Pre-FMM (continued)

We have

$$\sum_{n=1}^{K} M_n = M$$

Consider a uniform distribution, then

$$N_n \sim const \sim \frac{NPow(d)}{K},$$

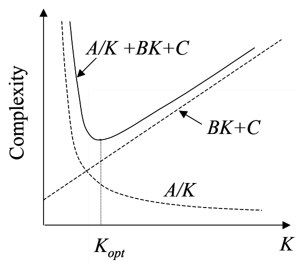
Power of the neighborhood of dimensionality d (the number of boxes in the neighborhood)

$$F(K) = \sum_{n=1}^{K}[(N-N_n)p + M_n p + M_n N_n] = KNp - NpPow(d) + Mp + \frac{MNPow(d)}{K}$$

$$= \frac{MN}{K}Pow(d) + (K - Pow(d))Np + Mp$$

$$Complexity = O(F(K))$$

Optimization of the box number



Optimum complexity

$$F(K) = \frac{MN}{K} Pow(d) + (K - Pow(d))Np + Mp$$

$$K_{opt} = \left[\frac{MNPow(d)}{Np}\right]^{1/2} = \sqrt{\frac{MPow(d)}{p}}.$$

$$Complexity = O(F(K_{opt})) = O\left(Np\left(2\sqrt{\frac{MPow(d)}{p}} - Pow(d)\right) + Mp\right)$$

For $M \sim N,\ p \ll N$:

$$Complexity = O\left(N^{3/2}p^{1/2}\right)$$

Actual complexity of "Pre-FMM"

Assume $M \sim N$ and $p \sim \log N$. Then complexity of the "Pre-FMM" is

$$C = O(p^{1/2}N^{3/2}) = O(N^{3/2}\log^{1/2}N).$$

For $p \sim \log\frac{1}{\epsilon}$ we have

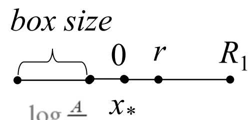
$$C = O(p^{1/2}N^{3/2}) = O(N^{3/2}\log^{1/2}\frac{1}{\epsilon}).$$

Optimize with error bound constraint

How the complexity changes, if we change the size of the neighborhood and request the same accuracy of the computation?

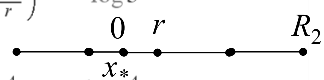
Complexity ($M \sim N \gg p$)

$$C \sim 2N^{3/2} p^{1/2} \sqrt{Pow(d)}\,.$$



1). $d = 1$, Neighborhoods of chess radius 1:

$$C_1 \sim 2N^{3/2} p_1^{1/2} \sqrt{3}\,, \quad p_1 \sim \frac{\log \frac{A}{|\epsilon|}}{\log\left(\frac{R_1}{r}\right)} = \frac{\log \frac{A}{|\epsilon|}}{\log 3}$$



2). $d = 1$, Neighborhoods of chess radius 2:

$$C_2 \sim 2N^{3/2} p_2^{1/2} \sqrt{5}\,, \quad p_2 \sim \frac{\log \frac{A}{|\epsilon|}}{\log\left(\frac{R_2}{r}\right)} = \frac{\log \frac{A}{|\epsilon|}}{\log 5}$$

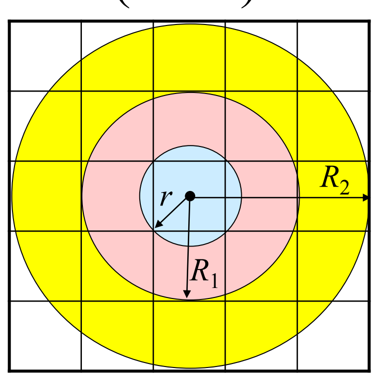
Then

~ 1.07

$$\frac{C_2}{C_1} \sim \frac{2N^{3/2} p_2^{1/2} \sqrt{5}}{2N^{3/2} p_1^{1/2} \sqrt{3}} = \sqrt{\frac{5p_2}{3p_1}} = \sqrt{\frac{5 \log 3}{3 \log 5}} = \sqrt{\frac{\log 243}{\log 125}} > 1.$$

Chess radius $= 1$ *is better!*

Optimize with error bound constraint ($d = 2$)



Chess radius $= 1$
is better!

$$\frac{C_2}{C_1} \sim \sqrt{\frac{25p_2}{9p_1}} = \frac{5}{3}\sqrt{\frac{p_2}{p_1}} = \frac{5}{3}\sqrt{\frac{\log\left(\frac{R_1}{r}\right)}{\log\left(\frac{R_2}{r}\right)}} = \frac{5}{3}\sqrt{\frac{\log\left(\frac{3}{\sqrt{2}}\right)}{\log\left(\frac{5}{\sqrt{2}}\right)}} \approx 1.29 > 1$$