
Lecture 13

Instructor: Carlos Pérez Arancibia

MIT
Mathematics Department
Cambridge, October 24, 2017
Consider matrices $A \in \mathbb{C}^{n \times n}$, where $n$ is a power of two: $n = 2^p$. We define the matrix format $\mathcal{H}_p$ recursively starting with the representation $\mathcal{H}_0$ which corresponds to scalars ($1 \times 1$ matrices). Assuming that the format $\mathcal{H}_{p-1}$ for matrices $\mathbb{C}^{2^{p-1} \times 2^{p-1}}$ is known, then a matrix $A \in \mathbb{C}^{2^p \times 2^p}$ can be represented as the following block matrix

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad A_{ij} \in \mathbb{C}^{2^{p-1} \times 2^{p-1}}.$$

We then restrict the set of all $A \in \mathbb{C}^{2^p \times 2^p}$ by the conditions

$$A_{11}, A_{22} \in \mathcal{H}_{p-1}, \quad A_{12}, A_{21} \in \mathbb{C}^{2^{p-1} \times 2^{p-1}}.$$
**Model format $\mathcal{H}_p$**

**Number of blocks in the $\mathcal{H}_p$ format.** Start with $p = 0$. The $1 \times 1$ matrix contains $N(0) = 1$ block. Now by the recursion we have $N(p) = 2 + 2N(p - 1)$ for $p > 0$. Therefore, we conclude that

$$N(p) = 3n - 2, \quad n = 2^p.$$  

**Storage cost.** The storage cost required for a matrix $A \in \mathbb{C}^{2^p \times 2^p}_k$ is $k \cdot 2^{p+1}$. Let $S(p)$ be the storage cost of a matrix in $\mathcal{H}_p$. For $p = 0$ we have $S(0) = 1$. The recursion formula shows that

$$S(p) = k \cdot 2^p + 2S(p - 1).$$

Therefore, solving the difference equation with initial condition $S(0) = 1$ we obtain $S(p) = (pk + 1)2^p = (k \log_2 n + 1)n$. 

Algebraic operations in $\mathcal{H}_p$

**Mat-vec product.** Let $N(p)$ denotes the cost of a matrix-vector multiplication $Ax$, where $A \in \mathcal{H}_p$. For $p \geq 1$ we write

\[
A = \begin{bmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{bmatrix}
\quad \text{and} \quad
x = \begin{bmatrix}
x_1 \\
x_2
\end{bmatrix},
\]

where $A_{ij} \in \mathbb{C}^{2^{p-1} \times 2^{p-1}}$ and $x_j \in \mathbb{C}^{2^{p-1}}$, $i, j = 1, 2$. Therefore the multiplication requires computing the products $y_{11} = A_{11}x_1$, $y_{12} = A_{12}x_2$, $y_{21} = A_{21}x_1$ and $y_{22} = A_{22}x_2$ and the sums $y_{11} + y_{12}$ and $y_{21} + y_{22}$. Since $A_{21}, A_{12} \in \mathbb{C}_k^{2^{p-1} \times 2^{p-1}}$ the cost of each multiplication is $(4k - 1)n/2 - k$, whereas one addition takes $n/2$ operations. Therefore

\[
N(p) = 2N(p - 1) + 2((4k - 1)n/2 - k + n/2) = 2N(p - 1) + 2k(2^{p+1} - 1).
\]

Solving the difference equation we obtain:

\[
N(p) = 4kn \log_2 n + 2k - 2kn.
\]
Matrix addition. We distinguish three types of additions:

1. $A + B \in \mathbb{C}_k^{2^p \times 2^p}$ (in the sense of formatted addition) where $A, B \in \mathbb{C}_k^{2^p \times 2^p}$ and cost $N_{R+R}(p)$.

2. $A + B \in \mathcal{H}_p$ where $A, B \in \mathcal{H}_p$ and cost $N_{H+H}(p)$.

3. $A + B \in \mathcal{H}_p$ where $A \in \mathcal{H}_p$ and $B \in \mathbb{C}_k^{2^p \times 2^p}$ with cost $N_{H+R}(p)$.
Algebraic operations in $\mathcal{H}_p$

1. $N_{R+R}$ is the cost of the formatted addition of two matrices in $\mathbb{C}_k^{2^p \times 2^p}$:

$$N_{R+R}(p) = \mathcal{O}(k^2 2^p) + \mathcal{O}(k^3).$$

2. Letting $\mathcal{R}_p = \mathbb{C}_k^{2^p \times 2^p}$ we have that the sum has the structure

$$\begin{bmatrix}
\mathcal{H}_{p-1} & \mathcal{R}_{p-1} \\
\mathcal{R}_{p-1} & \mathcal{H}_{p-1}
\end{bmatrix} +
\begin{bmatrix}
\mathcal{H}_{p-1} & \mathcal{R}_{p-1} \\
\mathcal{R}_{p-1} & \mathcal{H}_{p-1}
\end{bmatrix} =
\begin{bmatrix}
\mathcal{H}_{p-1} + \mathcal{H}_{p-1} & \mathcal{R}_{p-1} + \mathcal{R}_{p-1} \\
\mathcal{R}_{p-1} + \mathcal{R}_{p-1} & \mathcal{H}_{p-1} + \mathcal{H}_{p-1}
\end{bmatrix}.$$ 

Therefore

$$N_{H+H}(p) = 2N_{H+H}(p - 1) + Ak^2 2^p + Bk^3 \quad (A, B \text{ constants}).$$

Thus $N_{H+H} = (1 + Bk^3) 2^p + Ak^2 p 2^p - Bk^3 = (1 + Bk^3)n + Ak^2 n \log_2 n - Bk^3$.

3. The sum has the structure

$$\begin{bmatrix}
\mathcal{H}_{p-1} & \mathcal{R}_{p-1} \\
\mathcal{R}_{p-1} & \mathcal{H}_{p-1}
\end{bmatrix} +
\begin{bmatrix}
\mathcal{R}_{p-1} & \mathcal{R}_{p-1} \\
\mathcal{R}_{p-1} & \mathcal{R}_{p-1}
\end{bmatrix} =
\begin{bmatrix}
\mathcal{H}_{p-1} + \mathcal{R}_{p-1} & \mathcal{R}_{p-1} + \mathcal{R}_{p-1} \\
\mathcal{R}_{p-1} + \mathcal{R}_{p-1} & \mathcal{H}_{p-1} + \mathcal{R}_{p-1}
\end{bmatrix}.$$ 

Thus $N_{H+R} = (1 + Bk^3) 2^p + Ak^2 p 2^p - Bk^3 = (1 + Bk^3)n + Ak^2 n \log_2 n - Bk^3$. 
Matrix-matrix multiplication. Let $\mathbb{C}^{2^p \times 2^p} = \mathcal{R}_p$. We distinguish four cases

1. $A \cdot B \in \mathcal{R}_p$ for $A, B \in \mathcal{R}_p$ (cost $N_{R \cdot R}(p)$).

2. $A \cdot B \in \mathcal{R}_p$ for $A \in \mathcal{R}_p$ and $B \in \mathcal{H}_p$ (cost $N_{R \cdot H}(p)$).

3. $A \cdot B \in \mathcal{R}_p$ for $A \in \mathcal{H}_p$ and $B \in \mathcal{R}_p$ (cost $N_{H \cdot R}(p)$).

4. $A \cdot B \in \mathcal{H}_p$ for $A, B \in \mathcal{H}_p$ (cost $N_{H \cdot H}(p)$).
### Algebraic operations in $\mathcal{H}_p$

1. $N_{R \cdot R} = O(k^2 2^p) = O(k^2 n)$ (to produce the outer product form of the multiplication).

2. Let $A \in \mathcal{H}_p$ and $B \in \mathcal{R}_p$. We have $AUV^T = (AU)V^T$ where $AU \in \mathcal{R}_p$. $AU$ is equivalent to perform $k$ matrix-vector products which costs $O(kn \log_2 n)$ operations.

3. Let $A \in \mathcal{R}_p$ and $B \in \mathcal{H}_p$. Then $AB = UV^T B = U(V^T B) = U(B^T V)^T$. $B^T V \in \mathcal{R}_p$ requires $O(kn \log_2 n)$ operations.

4. Let $A \in \mathcal{H}_p$ and $B \in \mathcal{H}_p$. The product has the form

$$
\begin{bmatrix}
\mathcal{H}_{p-1}
&
\mathcal{R}_{p-1}

\mathcal{R}_{p-1}
&
\mathcal{H}_{p-1}
\end{bmatrix}
\begin{bmatrix}
\mathcal{H}_{p-1}
&
\mathcal{R}_{p-1}

\mathcal{R}_{p-1}
&
\mathcal{H}_{p-1}
\end{bmatrix}
= 
\begin{bmatrix}
\mathcal{H}_{p-1} \cdot \mathcal{H}_{p-1} + \mathcal{R}_{p-1} \cdot \mathcal{R}_{p-1} & \mathcal{H}_{p-1} \cdot \mathcal{R}_{p-1} + \mathcal{R}_{p-1} \cdot \mathcal{H}_{p-1}

\mathcal{R}_{p-1} \cdot \mathcal{H}_{p-1} + \mathcal{H}_{p-1} \cdot \mathcal{R}_{p-1} & \mathcal{R}_{p-1} \cdot \mathcal{R}_{p-1} + \mathcal{H}_{p-1} \cdot \mathcal{H}_{p-1}
\end{bmatrix}
$$

The cost is then given by

$$
N_{H \cdot H}(p) = 2N_{H \cdot H}(p-1) + 2N_{R \cdot R}(p-1) + N_{H \cdot R}(p-1) + N_{R \cdot H}(p-1) + 2N_{H + R}(p) + N_{R + R}(p)
$$

Thus $N_{H \cdot H}(p) = 2N_{H \cdot H}(p-1) + O(n \log_2 n) + O(n)$, which yields

$$
N_{H \cdot H}(p) = O(n \log_2^2 n) + O(n \log_2 n) + O(n)
$$

operations.
Matrix inversion. We want to approximate the inverse $A^{-1}$ of a matrix $A \in \mathcal{H}_p$. We define the inversion recursively. For $p = 0$ the inverse is exact. For $p \geq 1$ the exact inverse of $A$ has the block structure:

$$A^{-1} = \begin{bmatrix}
A_{11}^{-1} + A_{11}^{-1} A_{12} S^{-1} A_{21} A_{11}^{-1} & -A_{11}^{-1} A_{12} S^{-1} \\
-S^{-1} A_{21} A_{11}^{-1} & S^{-1}
\end{bmatrix}$$

involving the Schur complement $S = A_{22} - A_{21} A_{11}^{-1} A_{12}$. The cost of computing the approximate inverse $A^{-1}$ amounts to

$$N_{\text{inv}}(p) = 2N_{\text{inv}}(p-1) + 2N_{R \cdot H}(p-1) + 2N_{H \cdot R}(p-1) + 2N_{H + R}(p-1) + 2N_{R \cdot R}(p-1)$$

Therefore $N_{\text{inv}}(p) = O(n \log_2^2 n) + O(n \log_2 n)$. 
Example: 1D Green’s matrix

We already justified the separability of the off-diagonal blocks of the Laplace and Helmholtz Green’s functions in 2D. We now consider, from an algebraic viewpoint, the separability of the Green’s matrix corresponding to the discretization of a 1D Dirichlet boundary value using a 3-point finite difference stencil.

The finite-difference discretization of a 1D boundary value problem leads to the tridiagonal matrix:

\[
T = \begin{bmatrix}
\alpha_1 & \beta_1 \\
\gamma_1 & \alpha_2 & & \\
& \ddots & \ddots & \\
& & \ddots & \beta_{n-1} \\
& & & \gamma_{n-1} & \alpha_n
\end{bmatrix}.
\]

The inverse \( G = T^{-1} \) corresponds to the approximation of the Green’s function of the problem.
Example: 1D Green’s matrix

**Theorem.** Assume $T$ is tridiagonal and invertible. The $G = T^{-1}$ has rank-1 off-diagonal blocks.

**Proof.**

Corollary. Let $T$ be banded with bandwidth $2p + 1$, then $T^{-1}$ has rank-$p$ off-diagonal blocks.

**H-matrices**

We introduce a H-matrix algorithm for the fast matrix-vector multiplication involving matrices arising from the discretization of boundary integral equation.

We consider a matrix $M \in \mathbb{R}^{M \times N}$ given by

$$M = (k(x_i, y_j))_{i=1,j=1}^{M,N} \quad \text{with} \quad x_i, y_j \in [0, 1),$$

where except for the singularity at $x = y$, the kernel $k : \mathbb{R}^2 \to \mathbb{R}$ satisfies:

$$\left| \frac{\partial^p}{\partial y^p} k(x, y) \right| \leq \frac{C p!}{|x - y|^p}, \quad p \in \mathbb{N}.$$

**Example.** The kernel $k(x, y) = \log |x - y|$ satisfies the smoothness condition with $C = 1/p$.

**Assumption.** We assume that the points $x_i$ and $y_j$ are uniformly distributed and ordered: $x_1 \leq x_2 \cdots \leq x_N$ and $y_1 \leq y_2 \cdots \leq y_M$. 
Admissible blocks

Let $I = \{1, \ldots, M\}$ and $J = \{1, \ldots, N\}$ be two index sets and let $X = \{x_i : i \in I\}$ and $Y = \{y_j : j \in J\}$.

Let $\mathcal{P}(I)$ (resp. $\mathcal{P}(J)$) denote a partition of $I$ (resp. $J$). Then for $\sigma \in \mathcal{P}(I)$ and $\tau \in \mathcal{P}(J)$ we defined

$$X(\sigma) = \{x_i \in X : i \in \sigma\},$$
$$Y(\tau) = \{y_j \in Y : j \in \tau\}.$$

We call $b = \sigma \times \tau$ a block of indices. Thus, for any block of indices $b = \tau \times \sigma$ we define the matrix block

$$M^b = (m_{ij})_{i \in \sigma, j \in \tau}.$$

**Admissible blocks.** Blocks of $M$ that admit a low-rank approximation.
Admissible blocks

The block partition of $M$ is based on geometrical considerations. It is useful then to define the radius $r_\sigma$ (resp. $r_\tau$) and the center $c_\sigma$ (resp. $c_\tau$) of $X(\sigma)$ (resp. $Y(\tau)$) as:

$$|x_i - c_\sigma| \leq r_\sigma, \quad i \in \sigma \quad \text{ (resp. } |y_j - c_\tau| \leq r_\tau, \quad j \in \tau).$$

**Definition.** A matrix block $M^b$, $b = \sigma \times \tau$, of $M$ is called admissible if there exists $\eta \in (0, 1]$ such that

$$\eta \text{ dist}(\sigma, \tau) = \eta \min_{i \in \sigma, j \in \tau} |x_i - y_j| \geq r_\tau + r_\sigma.$$
Hierarchical splitting

In order to split the matrix $M$ into admissible blocks we use a binary hierarchical splitting of the index sets $I$ and $J$.

We form a binary tree $T_I(\ell)$ where at level $\ell = 0$ we have $T_I(0) = I$. Then, the tree is built recursively dyadic partitioning $I$ until each leaf index set contains only a small number $\nu > 0$ of indices. At level $\ell > 0$ the nodes of the tree are given by the index sets

$$\sigma(\ell, m) = \{ i \in I : x_i \in [m/2^\ell, (m + 1)/2^\ell) \}, \quad m = 0, \ldots, 2^\ell - 1.$$ 

Similarly we obtain the tree $T_J(\ell)$ for $J$.

Note that at level $\ell$:

$$r_\sigma = \frac{1}{2^{\ell+1}} \quad \text{and} \quad c_\sigma = \frac{m + 1/2}{2^\ell}, \quad m = 0, \ldots, 2^\ell - 1.$$ 

We let $T_{I \times J} = T_I \times T_J$ denote the tensor block partition of $I \times J$. 
Hierarchical splitting into admissible blocks

- Split $M$ with respect to the blocks $b = \sigma \times \tau \in T_{I \times J}(2)$ and sort admissible and non-admissible blocks at level $\ell = 2$:

$$M = M_2 + N_2$$

where $M_2$ contains the admissible blocks of $T_{I \times J}(2)$ and $N_2$ contains the non-admissible blocks of $T_{I \times J}(2)$.

- Proceed with $N_2$ :

$$N_2 = M_3 + N_3$$

where $M_3$ contains the admissible blocks of $T_{I \times J}(3)$ contained in $N_2$ and $N_3$ contains the non-admissible blocks of $T_{I \times J}(3)$.

- Repeat this process up to level $L$ where we obtain

$$M = \sum_{\ell=2}^{L} M_\ell + N_L,$$

where the matrix $\sum_{\ell}^{L} M_\ell$ contains the admissible blocks at all levels. $\sum_{\ell}^{L} M_\ell$ is called far-field and $N_L$ is called near-field.
Hierarchical splitting into admissible blocks

\[ M = M_2 + N_2 \]
Hierarchical splitting into admissible blocks

\[ N_2 = M_3 + N_3 \]

(\( \eta = 1 \))
Hierarchical splitting into admissible blocks

\[ N_3 = M_4 + N_4 \]

(\( \eta = 1 \))
Hierarchical splitting into admissible blocks

\[ M = \sum_{\ell=2}^{4} M_\ell + N_4 \]
Let \( b = \sigma \times \tau \) be an admissible block, and let \( x \in X(\sigma) \) and \( y \in Y(\tau) \). Therefore:

\[
K(x, y) = \sum_{\ell=0}^{p-1} \frac{1}{\ell!} (y - c_{\tau})^{\ell} \frac{\partial^\ell}{\partial y^\ell} k(x, c_{\tau}) + R_p(x, y)
\]

\[
= \sum_{\ell=0}^{p-1} \frac{1}{\ell!} \varphi_{\ell}^{\sigma, \tau}(x) \psi_{\ell}^{\tau}(y) + R_p(x, y)
\]

where

\[
\varphi_{\ell}^{\sigma, \tau}(x) = \frac{\partial^\ell}{\partial y^\ell} k(x, c_{\tau}), \quad \psi_{\ell}^{\tau}(y) = (y - c_{\tau})^{\ell},
\]

and

\[
R_p(x, y) = \frac{1}{p!} (y - c_{\tau})^{p} \frac{\partial^p}{\partial y^p} k(x, \tilde{y}), \quad \tilde{y} = c_{\tau} + t(y - c_{\tau}), \; t \in (0, 1).
\]

---

**Low-rank approximation of admissible blocks**
Low-rank approximation of admissible blocks

Since the kernel satisfies

\[ \left| \frac{\partial^p}{\partial y^p} k(x, y) \right| \leq \frac{C p!}{|x - y|^p}, \quad p \in \mathbb{N}, \]

and

\[ \eta |x - \tilde{y}| \geq \eta \text{dist}(\sigma, \tau) \geq r_\tau + r_\sigma, \]

we easily obtain

\[ |R_p(x, y)| \leq C \eta^p \left( \frac{r_\tau}{r_\sigma + r_\tau} \right)^p. \]

Since \( 0 < \eta \leq 1 \) and \( r_\tau / (r_\sigma + r_\tau) = 1/2 \) we have that the error decreases exponentially as \( p \) increases. Therefore, \( M^b \) admits a low rank approximation:

\[ \tilde{M}^b = (\Phi^{\sigma, \tau})^T D \Psi^\tau, \]

where letting \( P = \{0, \ldots, p - 1\} \), \( D \in \mathbb{R}^{p \times p} \), \( \Phi^{\sigma, \tau} \in \mathbb{R}^{p \times \# \sigma} \) and \( \Psi^\tau \in \mathbb{R}^{p \times \# \tau} \) are given by

\[ D = \text{diag}(1/\ell!)_{\ell \in P}, \quad \Phi^{\sigma, \tau} = (\varphi^\sigma_\ell(x_i))_{\ell \in P, i \in \sigma} \quad \text{and} \quad \Psi^\tau = (\psi^\sigma_\ell(y_j))_{\ell \in P, j \in \tau}. \]
The matrix-vector multiplication can be computed approximately by

\[ f = Mx \approx \sum_{\ell=2}^{L} \tilde{M}_\ell x + N_L x \]

where the blocks \( \tilde{M}^b \) of \( \tilde{M}_\ell \) are low-rank matrices of rank \( p \).

- The multiplication of a block \( \tilde{M}^b \), \( b = \sigma \times \tau \) by a vector requires \( O(p(#\sigma + #\tau)) \) operations, where \( #\tau \leq M/2^\ell \) and \( #\sigma \leq N/2^\ell \). Since at level \( \ell \geq 2 \) there are \( 2^\ell \) blocks in \( \tilde{M}_\ell \) and only few of them (say \( \gamma > 0 \)) are non-zero, the computation of \( \tilde{M}_\ell x \) requires \( O(p(M + N)) \) operations. That is \( O(\gamma 2^\ell (M + N)/2^\ell) \) operations. For example \( \gamma = 2 \) if \( \eta = 1 \), and \( \gamma = 1 \) if \( \eta = 1/2 \).
There are $L$ matrices $\tilde{M}_\ell$, where $\nu 2^L = \max\{M, N\}$. Therefore, $L = \log_2(\max\{N + M\}/\nu)$. Thus the cost of computing $\sum_{\ell=2}^L \tilde{M}_\ell x$ is

$$O(p(M + N) \log_2(\max\{M, N\}/\nu)) = O(p(M + N) \log N)$$

if $O(N/M) = O(1)$.

$N_L$ is a $2^L \times 2^L$ block matrix. A few number of blocks (say $\beta \approx \eta^{-1} + 1 > 0$) per row are non-zero. Each non-zero block is a $\nu$ (or smaller) matrix. Therefore, the computation of $N_L x$ requires

$$O(2^L \beta \nu^2) = O(\max\{N, M\} \nu \beta) = O(N \nu \beta)$$

operations if $O(N/M) = O(1)$.

The total cost amounts to $O(p(N + M) \log N)$ operations.
FMM and $H$-matrices

Assume further that the kernel satisfies

$$\left| \frac{\partial^\beta}{\partial x^\beta} \frac{\partial^\gamma}{\partial y^\gamma} k(x, y) \right| \leq \frac{Cp!}{|x - y|^p}, \quad \gamma + \beta = p.$$ 

Therefore, theta Taylor expansion of $k$ around $(c_\sigma, c_\tau)$ yields

$$k(x, y) = \sum_{0 \leq \ell + m \leq p} \frac{1}{\ell!m!} \frac{\partial^\ell}{\partial x^\ell} \frac{\partial^m}{\partial y^m} k(c_\sigma, c_\tau) \varphi_\ell(x) \psi_m(y) + R_p(x, y)$$

where

$$\varphi_\ell(x) = (x - c_\sigma)\ell \quad \text{and} \quad \psi_m(y) = (y - c_\tau)^m.$$ 

Therefore

$$|R_p(x, y)| \leq \tilde{C} \frac{(|x - c_\sigma| + |y - c_\tau|^p)^p}{|\tilde{x} - \tilde{y}|^p},$$

where $\tilde{x} = c_\sigma + \theta_x(x - c_\sigma)$ and $\tilde{y} = c_\tau + \theta_y(y - c_\tau)$, $\theta_x, \theta_y \in (0, 1)$. 
Low-rank approximation of admissible blocks

If \( x \in X(\sigma) \), \( y \in Y(\tau) \) and \( b = \sigma \times \tau \) is an admissible index block, by the admissibility condition we have

\[
\eta |\tilde{x} - \tilde{y}| \geq \eta \text{dist}(\sigma, \tau) \geq r_\sigma + r_\tau
\]

with \( |x - c_\sigma| \leq c_\sigma \) and \( |y - c_\tau| \leq r_\tau \). Therefore, we conclude that

\[
|R_p(x, y)| \leq C \eta^p.
\]

If \( \eta < 1 \), then the error in the low rank approximation

\[
\widehat{M}^b = (\Phi^\sigma)^T A^{\sigma, \tau} \Psi^\tau,
\]

converges exponentially fast as \( p \) increases, where \( \Phi^\sigma = (\varphi^\sigma(x_i))_{i \in \sigma} \in \mathbb{R}^{p \times \#\sigma} \), \( \Psi^\tau = (\psi^\tau(y_j))_{j \in \tau} \in \mathbb{R}^{p \times \#\tau} \) and \( A^{\sigma, \tau} = (a^{\sigma, \tau}_{\ell, m})_{\ell, m \in P} \in \mathbb{R}^{p \times p} \), with

\[
a^{\sigma, \tau}_{\ell, m} = \begin{cases} 
\frac{1}{\ell! m!} \frac{\partial^{\ell}}{\partial x^{\ell}} \frac{\partial^m}{\partial y^m} k(c_\sigma, c_\tau) & \text{if} \quad 0 \leq \ell + m \leq p - 1, \\
0 & \text{otherwise}.
\end{cases}
\]
Given that each block \( \tilde{M}^b \) of \( \tilde{M}_\ell \) is a low-rank matrix, we write

\[
\tilde{M}_\ell = \text{blockdiag}(\Phi^\sigma)^T_{\sigma \in T_J(\ell)} A_\ell \text{ diag}(\Psi^\tau)_{\tau \in T_I(\ell)}
\]

where \( A_\ell \in \mathbb{R}^{p^{2\ell} \times p^{2\ell}} \) has non-zero blocks \( A^{\sigma,\tau} \in \mathbb{R}^{p \times p} \) at the position of the non-zero blocks of \( \tilde{M}_\ell \).

For example:

\[
\tilde{M}_2 = \begin{array}{cc}
M_{13} & M_{14} \\
M_{24} & \\
\end{array}
\begin{array}{cc}
\end{array}
\begin{array}{c}
M_{31} \\
\end{array}
\begin{array}{c}
M_{41}\\
M_{42}
\end{array}
\]

for \( \ell = 2 \) and \( \eta = 1 \).
Given that each block $\tilde{M}_b^\ell$ of $\tilde{M}_\ell$ is a low-rank matrix, we write

$$\tilde{M}_\ell = \text{blockdiag}(\Phi_{\sigma})^T_{\sigma \in T_J(\ell)} A_\ell \text{ diag}(\Psi^\tau)_{\tau \in T_I(\ell)}$$

where $A_\ell \in \mathbb{R}^{p^{2\ell} \times p^{2\ell}}$ has non-zero blocks $A^{\sigma,\tau} \in \mathbb{R}^{p \times p}$ at the position of the non-zero blocks of $\tilde{M}_\ell$.

For example:

$$\tilde{M}_2 = \begin{pmatrix}
(\Phi_1)^T & \cdot & \cdot & \cdot \\
\cdot & (\Phi_2)^T & \cdot & \cdot \\
\cdot & \cdot & (\Phi_3)^T & \cdot \\
\cdot & \cdot & \cdot & (\Phi_4)^T \\
\end{pmatrix}$$

$$\begin{pmatrix}
A_{13} & A_{14} \\
\cdot & A_{24} \\
A_{31} & \cdot \\
A_{41} & A_{42} \\
\end{pmatrix}$$

$$\begin{pmatrix}
\Psi_1 \\
\Psi_2 \\
\Psi_3 \\
\Psi_4 \\
\end{pmatrix}$$

for $\ell = 2$ and $\eta = 1$. 
We need the matrices to satisfy a consistency condition:

\( \sigma', \sigma'' \in T_I(\ell + 1) \) are children of \( \sigma \in T_I(\ell) \)

\( \tau', \tau'' \in T_J(\ell + 1) \) are children of \( \tau \in T_J(\ell) \),

then

\[
\Phi^\sigma = \begin{bmatrix}
C^{\sigma,\sigma'} & C^{\sigma,\sigma''}
\end{bmatrix}
\begin{bmatrix}
\Phi^\sigma' \\
\Phi^\sigma''
\end{bmatrix}
\]

and

\[
\Psi^\tau = \begin{bmatrix}
C^{\tau,\tau'} & C^{\tau,\tau''}
\end{bmatrix}
\begin{bmatrix}
\Psi^\tau' \\
\Psi^\tau''
\end{bmatrix}
\]

The consistency condition resembles the projection rule in the FMM!
We note that
\begin{equation}
\varphi_{\ell}^{\sigma}(x) = (x - x_{\sigma})^\ell = ((x - c_{\sigma}) - (c_{\sigma} - c_{\sigma'}))^\ell
\end{equation}

\begin{equation}
= \sum_{m=0}^{\ell} \binom{\ell}{m} (c_{\sigma'} - c_{\sigma})^{\ell-m} (x - c_{\sigma})^m
\end{equation}

for all \( \ell = 0, \ldots, p - 1 \). We thus have

\begin{equation}
((x_i - c_{\sigma})^\ell)_{i \in \sigma'} = \sum_{m=0}^{\ell} c_{\ell,m}^{\sigma,\sigma'} ((x_i - c_{\sigma'})^m)_{i \in \sigma'}
\end{equation}

from where we identify

\begin{equation}
C^{\sigma,\sigma'} = (c_{\ell,m}^{\sigma,\sigma'})_{\ell,m \in P}
\end{equation}

where

\begin{equation}
c_{\ell,m}^{\sigma,\sigma'} = \begin{cases} 
0 & \ell < m \\
\binom{\ell}{m} (c_{\sigma} - c_{\sigma'})^{\ell-m} & \ell \geq m.
\end{cases}
\end{equation}
For $\ell = 2, \ldots, n - 1$ let

$$D_{\ell,\ell+1}^{\Phi} = \text{blockdiag}(\begin{bmatrix} C^{\sigma,\sigma'} & C^{\sigma,\sigma''} \end{bmatrix})_{\sigma \in T_I(\ell)} \in \mathbb{R}^{p2^\ell \times 2p2^\ell}.$$ 

Therefore, the consistency condition at level $\ell$ is given by

$$\text{blockdiag} (\Phi^\sigma)_{\sigma \in T_I(\ell)} = D_{\ell,\ell+1}^{\Phi} \text{blockdiag} (\Phi^\sigma)_{\sigma \in T_I(\ell+1)}.$$ 

Recursive application of the consistency condition leads to

$$\widehat{M}_\ell = \text{blockdiag}(\Phi^\sigma)^T_{\sigma \in T_I(\ell+1)} (D_{\ell,\ell+1}^{\Phi})^T A_\ell D_{\ell,\ell+1}^{\Psi} \text{blockading}(\Psi^\tau)_{\tau \in T_J(\ell+1)}$$

$$= \text{blockading}(\Phi^\sigma)^T_{\sigma \in T_I(L)} (D_{L-1,L}^{\Phi})^T \cdots (D_{\ell,\ell+1}^{\Phi})^T A_\ell \times$$

$$\times D_{\ell,\ell+1}^{\Psi} \cdots D_{L-1,L}^{\Psi} \text{blockading}(\Psi^\tau)_{\tau \in T_J(L)}.$$ 

**Observation.** The factors $\text{blockdiag}(\Phi^\sigma)^T_{\sigma \in T_I(L)}$ and $\text{blockdiag}(\Psi^\tau)_{\tau \in T_J(L)}$ appear in all matrices $\widehat{M}_\ell$ ($\ell = 2, \ldots, n$) and the factors $D_{i,i+1}^{\Phi}$ and $D_{i,i+1}^{\Psi}$ appear in all matrices $\widehat{M}_\ell$ with $\ell \leq i$. 
**O(N) algorithm**

- **Initialization:**
  \[ x_L = \text{blockdiag}(\Psi^\tau)_{\tau \in T_J(L)} x \in \mathbb{R}^{p2^L}. \]

  Complexity: The matrix \( \text{blockdiag}(\Psi^\tau)_{\tau \in T_J(L)} \) contains \( 2^L = N/\nu \) non-zero blocks of size \( p \times \nu \). Then this matrix-vector product requires \( O(pN) \) operations.

  For \( \ell = L - 1, \ldots, 2 \) compute
  \[ x_\ell = D_{\ell,\ell+1}^\Psi x_{\ell+1}. \]

  Complexity: \( D_{\ell,\ell+1}^\Psi \) consists of \( 2^\ell \) non-zero blocks of the form \([C^{\sigma',\sigma'} C^{\sigma,\sigma''}] \in \mathbb{R}^{p \times 2p}\). We thus require less than \( 2p^22^\ell \) operations in step \( \ell \). Hence the computation cost is \( \sum_{\ell=2}^L O(2p^22^\ell) = O(2p^22^L) = O(2p^2N/\nu) \).

- (Upward pass) For \( \ell = 2, \ldots, L \) compute
  \[ y_\ell = A_\ell x_\ell. \]

  Complexity: There at most \( 2^\ell \gamma \) non-zero blocks of \( A_\ell \) at level \( \ell \) and each block of \( A_\ell \) is of size \( p \times p \). Thus \( A_\ell x_\ell \) requires \( O(p^22^\ell) \) operations. The overall const is then \( \sum_{\ell=2}^L O(p^22^\ell) = O(p^22^L) = O(p^2N/\nu) \).
(Downward pass) Compute

\[ f_F = \sum_{\ell=2}^{L-1} \text{blockdiag}(\Phi^\sigma)^T_{\sigma \in \mathcal{T}_I(L)} (D^\Phi_{L-1,L})^T \cdots (D^\Phi_{\ell,\ell+1})^T y_\ell. \]

We apply Horner’s rule. Set \( z_2 = y_2 \) and compute

\[ z_\ell = (D^\Phi_{\ell-1,\ell})^T z_{\ell-1} + y_\ell \]

for \( \ell = 3, \ldots, L \).

Complexity: The multiplication by \( (D^\Phi_{\ell,\ell+1})^T \) requires \( O(2p^22^\ell) \) operations. Hence, in total this takes \( O(p^2N/\nu) \) operations.

Then

\[ f_F = \text{blockdiag}(\Phi^\sigma)^T_{\sigma \in \mathcal{T}_I(L)} z_L. \]

Complexity: \( O(pM) \).

(Near-field). Compute \( f_N = N_L x \) directly and add it to \( f_F \).

Complexity: \( O(N\nu\beta) \) (\( \beta \approx 1/\eta + 1 \)) operations.
Selecting $\nu = p$, the total computational cost amounts to

$$O(p(N + M)) = O(N + M).$$