$\mathcal{H}^2$ Matrix Compression

Srinath Kailasa [*]

University College London

November 26, 2022

This note summarises the key results from Steffen Böerm's book [1].

0.1 Chapter 1 - Introduction

The fundamental iea is to reduce storage requirements using an alternative multilevel representation of a dense matrix instead of the standard representation by a 2D array.

Hierarchical matrices are the algebraic counterparts of panel clustering and multipole methods. A partition of the matrix takes the place of the partition of the domains of interaction, and low-rank submatrices take the place of local separable expansions. Due to their algebraic structure, h-matrices can be applied to more general problems than just integral equations.

Construction of a cluster basis, a multilevel basis, is the key challenge of $\mathcal{H}^2$. Questions:

1. What types of matrices can we actually compress?

2. What problems can be efficiently solved by $\mathcal{H}^2$ matrices?

To (1) we will find that generally integral operators which exhibit asymptotically smooth kernel functions can be compressed. There are also general techniques to estimate whether a given matrix can be placed in this framework. Including the solution operators of ODEs/PDEs. The simplest case is that of an integral operator which can be replaced by a separable approximation (e.g. multipole style).

[*] srinath.kailasa.18@ucl.ac.uk

For the latter the standard approach is to use cluster trees. I.e. to split the domains defining the integral operator into a hierarchy of subdomains and use an efficient recursive scheme to find an almost optimal decomposition of the original integral operator into local operators which can be approximated.

We consider 3 types of efficient operations.

1. The construction of an approximation of the system matrix.
2. Arithmetic operations like matrix-vector and matrix-matrix products.
3. Complex operations like matrix factorisations and matrix inversion, constructed out of basic arithmetic operations.

Construction

An $\mathcal{H}^2$ can be constructed in several ways: if it's the approximation of an explicitly given integral operator we can compute the $\mathcal{H}^2$ my discretising a number of local separable approximations (e.g. like a multipole expansion). If we want to approximate a given matrix we can use compression algorithms, but these are quasi-optimal approximations. But still useful, as then we can use $\mathcal{H}^2$ as a black box method. These techniques can be combined too in a general and simple interpolation scheme.

Arithmetic

If we want to solve a system of linear equations with a system in $\mathcal{H}^2$ representation, we at least have to be able to evaluate matvecs. This and related ops, like the product with the transposed matrix or forward/backward substitution steps can be accomplished in optimal complexity, i.e. no more than two operations required per unit of storage (?).

Inversion/Preconditioners

Using matrix-matrix products we can perform more complicated arithmetic like LU factorisations or inversions.

Which problems can be solved efficiently

The book focuses on discretisations of integral equations, especially those for homogenous elliptic pdes with the boundary integral method. Also consider the construction of approximate inverses for stiffness matrices arising from FEM, as they've been proven to be well approximated by $\mathcal{H}^2$

References

[1] Steffen Börm. *Efficient numerical methods for non-local operators: H2-matrix compression, algorithms and analysis*. Vol. 14. European Mathematical Society, 2010.