

# Improving Matrix-Matrix Multiplication in Algebraic Multigrid Context

Majid Rasouli  
University of Utah  
Salt Lake City, Utah  
rasouli@cs.utah.edu

Hari Sundar  
School of Computing, University of Utah  
Salt Lake City, Utah  
hari@cs.utah.edu

## ABSTRACT

This paper provides a new method for matrix-matrix multiplication.

## KEYWORDS

Algebraic Multigrid, AMG, Sparse, Computational Linear Algebra

### ACM Reference Format:

Majid Rasouli and Hari Sundar. 2019. Improving Matrix-Matrix Multiplication in Algebraic Multigrid Context. In *Proceedings of ACM Symposium on High-Performance Parallel and Distributed Computing (HPDC'19)*. ACM, New York, NY, USA, 1 page. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## 1 INTRODUCTION

This is intro.

## 2 METHODS

To compute the coarse matrix  $Ac$ , a triple matrix multiplication should be done:

$$Ac = R \times A \times P \quad (1)$$

in which  $R = P^T$ . We do it in two parts, performing matrix-matrix multiplications (MATMULT) twice: first  $A * P$ , then  $R * (A * P)$ .

The matrices are partitioned on multiple processors by row blocks (Figure 1). Matrices  $A$  and  $P$  have the same number of rows and consequently are partitioned the same way.  $R$  has less number of rows and has a different partition.

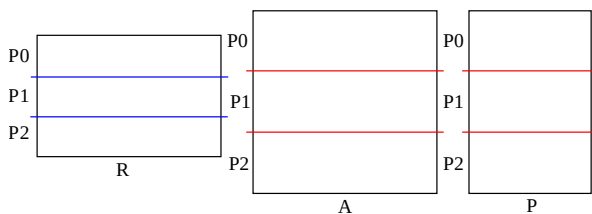


Figure 1

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

HPDC'19, June 2019, Phoenix, Arizona USA

© 2019 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## 2.1 Part1

## 2.2 Part2

## 3 NUMERICAL RESULTS

This is results.