Communication-Optimal Loop Nests

Nicholas Knight

New York University
Dept. of Mathematics and
Center for Data Science
nknight@nyu.edu

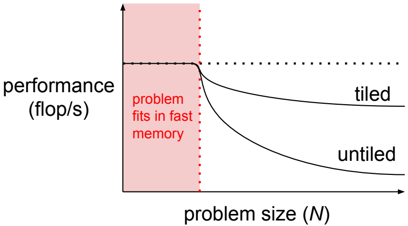
SIAM Annual Meeting
July 13, 2017

Joint work with:

- Michael Christ, UC-Berkeley, Dept. of Mathematics
- James Demmel, UC-Berkeley, Depts. of Mathematics and EECS
- Thomas Scanlon, UC-Berkeley, Dept. of Mathematics
- Katherine Yelick, Lawrence Berkeley Natl. Lab. and UC-Berkeley, Dept. of EECS

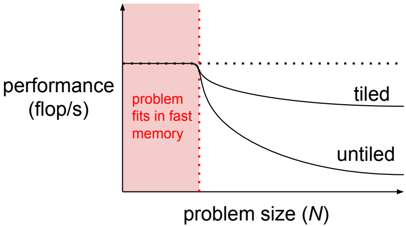
Modeling Communication Cost

Matrix Multiplication
(tiled vs. untiled implementation)



Modeling Communication Cost

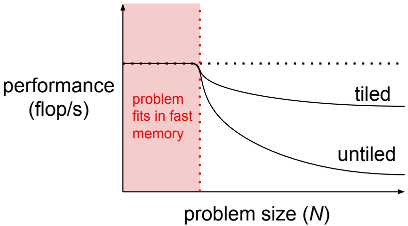
Matrix Multiplication
(tiled vs. untiled implementation)



Observation: # operations is generally unreliable for predicting runtime — neglects the cost of communication.

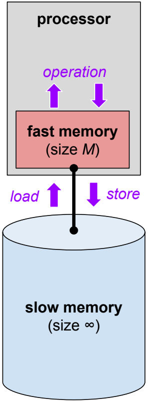
Modeling Communication Cost

Matrix Multiplication
(tiled vs. untiled implementation)



Observation: $\#$ operations is generally unreliable for predicting runtime — neglects the cost of communication.

Model Machine:



Communication Cost $\mathcal{C}$

$$\mathcal{C} = \#\,\text{loads/stores}$$

Motivating Application: Matrix Multiplication

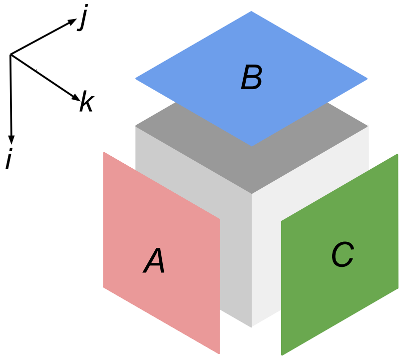
Matrix multiplication algorithm:

for $i = 1 : N$
  for $j = 1 : N$
    for $k = 1 : N$
      $\mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}$

Motivating Application: Matrix Multiplication

Matrix multiplication algorithm:

$$
\begin{array}{l}
\text{for } i = 1 : N \\
\quad \text{for } j = 1 : N \\
\quad\quad \text{for } k = 1 : N \\
\quad\quad\quad \mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}
\end{array}
$$



Geometric interpretation:

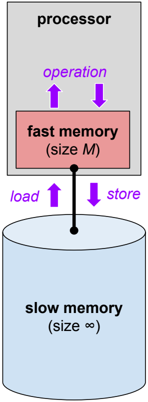
- Operations are points (i, j, k) in Euclidean space;
- Operands are projections (i, k), (k, j), (i, j) onto the coordinate planes.

Matrix Multiplication: Untiled Implementation

Untiled implementation:

for $i = 1 : N$
    for $j = 1 : N$
        for $k = 1 : N$
            load $\mathbf{A}_{i,k}, \mathbf{B}_{k,j}, \mathbf{C}_{i,j}$
            $\mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}$
            store $\mathbf{C}_{i,j}$

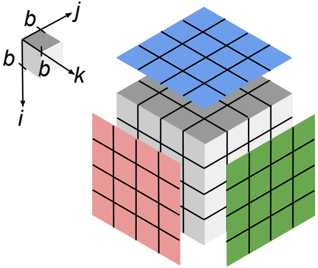
$\mathcal{C} \approx N^3$ (no data reuse)



Matrix Multiplication: Tiled Implementation

Tiled implementation:

for $\hat{\imath} = 1 : b : N$
  for $\hat{\jmath} = 1 : b : N$
    for $\hat{k} = 1 : b : N$
      load blocks $\mathbf{A}_{\hat{\imath},\hat{k}}, \mathbf{B}_{\hat{k},\hat{\jmath}}, \mathbf{C}_{\hat{\imath},\hat{\jmath}}$
      for $i = \hat{\imath} : \hat{\imath} + b - 1$
        for $j = \hat{\jmath} : \hat{\jmath} + b - 1$
          for $k = \hat{k} : \hat{k} + b - 1$
            $\mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}$
      store block $\mathbf{C}_{\hat{\imath},\hat{\jmath}}$



$$\mathcal{C} \approx \frac{N^3}{b} \quad (b\text{-fold data reuse})$$

($b = 1$: untiled impl.)

Matrix Multiplication: Tiling Attains Lower Bound

Picking tiling parameter b to minimize communication cost $\mathcal{C}$

- b-fold reuse means we want to maximize b.
- **A**-, **B**-, **C**-blocks must fit in fast memory ($3b^2 \leq M$).

$$\text{Picking } \; b \approx M^{1/2} \qquad \Rightarrow \qquad \mathcal{C} \approx \frac{N^3}{M^{1/2}}.$$

Matrix Multiplication: Tiling Attains Lower Bound

Picking tiling parameter b to minimize communication cost $\mathcal{C}$

- b-fold reuse means we want to maximize b.
- **A**-, **B**-, **C**-blocks must fit in fast memory ($3b^2 \leq M$).

$$\text{Picking} \;\; b \approx M^{1/2} \qquad \Rightarrow \qquad \mathcal{C} \approx \frac{N^3}{M^{1/2}}.$$

Communication Lower Bound (Hong-Kung, 1981)

In any implementation of matrix multiplication, $\mathcal{C} \succeq \dfrac{N^3}{M^{1/2}}$.

Matrix Multiplication: Tiling Attains Lower Bound

Picking tiling parameter b to minimize communication cost $\mathcal{C}$

- b-fold reuse means we want to maximize b.
- **A**-, **B**-, **C**-blocks must fit in fast memory ($3b^2 \leq M$).

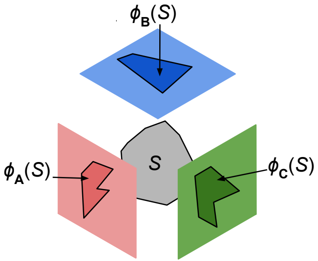
$$\text{Picking } \; b \approx M^{1/2} \qquad \Rightarrow \qquad \mathcal{C} \approx \frac{N^3}{M^{1/2}}.$$

Communication Lower Bound (Hong-Kung, 1981)

In any implementation of matrix multiplication, $\mathcal{C} \succeq \dfrac{N^3}{M^{1/2}}$.

- *Iteration space tiling*: heuristic to reduce $\mathcal{C}$ in loop nests.
- This loop nest: tiling with b-by-b-by-b cubes minimizes $\mathcal{C}$.

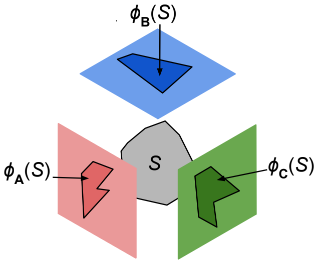
Lower Bound Ingredient #1: Loomis-Whitney Inequality



Array subscripts are linear maps from $\mathbb{Z}^3$ to $\mathbb{Z}^2$:

$$\phi_{\mathbf{A}} : (i, j, k) \mapsto (i, k),$$
$$\phi_{\mathbf{B}} : (i, j, k) \mapsto (k, j),$$
$$\phi_{\mathbf{C}} : (i, j, k) \mapsto (i, j).$$

Lower Bound Ingredient #1: Loomis-Whitney Inequality



Loomis-Whitney Inequality

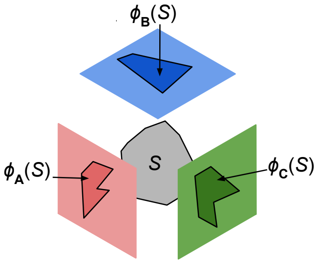
For any $S \subseteq \mathbb{Z}^3$,

$$|S| \leq |\phi_{\mathbf{A}}(S)|^{1/2} \cdot |\phi_{\mathbf{B}}(S)|^{1/2} \cdot |\phi_{\mathbf{C}}(S)|^{1/2}.$$

Array subscripts are linear maps from $\mathbb{Z}^3$ to $\mathbb{Z}^2$:

$$\phi_{\mathbf{A}} : (i,j,k) \mapsto (i,k),$$
$$\phi_{\mathbf{B}} : (i,j,k) \mapsto (k,j),$$
$$\phi_{\mathbf{C}} : (i,j,k) \mapsto (i,j).$$

Lower Bound Ingredient #1: Loomis-Whitney Inequality



Array subscripts are linear maps from $\mathbb{Z}^3$ to $\mathbb{Z}^2$:

$$\phi_{\mathbf{A}} : (i,j,k) \mapsto (i,k),$$
$$\phi_{\mathbf{B}} : (i,j,k) \mapsto (k,j),$$
$$\phi_{\mathbf{C}} : (i,j,k) \mapsto (i,j).$$

Loomis-Whitney Inequality

For any $S \subseteq \mathbb{Z}^3$,

$$|S| \leq |\phi_{\mathbf{A}}(S)|^{1/2} \cdot |\phi_{\mathbf{B}}(S)|^{1/2} \cdot |\phi_{\mathbf{C}}(S)|^{1/2}.$$

The *memory footprint* of iterations S is

$$\mu(S) = |\phi_{\mathbf{A}}(S)| + |\phi_{\mathbf{B}}(S)| + |\phi_{\mathbf{C}}(S)|.$$

Loomis-Whitney gives the lower bound,

$$\mu(S) \geq |S|^{2/3}.$$

Lower Bound Ingredient #2: Segmentation

⋮
operation
load
store
operation
load
load
operation
store
store
load
operation
store
load
⋮

$$\left.\begin{array}{l}\text{operation}\\\text{load}\\\text{load}\\\text{operation}\\\text{store}\\\text{store}\end{array}\right\}\begin{array}{l}\text{segment } i\\\text{iterations } S_i\\|S_i| = 2\end{array}$$

Lower Bound Ingredient #2: Segmentation

$\vdots$
operation
load
store
operation
load
load
operation
store
store
load
operation
store
load
$\vdots$

segment i
iterations S_i
$|S_i| = 2$

For any segment i,

$$\begin{aligned} \mathcal{C}_i &\geq \mu(S_i) - 2M \\ &\geq |S_i|^{2/3} - 2M. \end{aligned}$$

Lower Bound Ingredient #2: Segmentation

⋮
operation
load
store
operation
load
load
operation
store
store
load
operation
store
load
⋮

segment i
iterations S_i
$|S_i| = 2$

For any segment i,

$$\begin{aligned} \mathcal{C}_i &\geq \mu(S_i) - 2M \\ &\geq |S_i|^{2/3} - 2M. \end{aligned}$$

Summing over all segments,

$$\mathcal{C} \geq \sum_i |S_i|^{2/3} - 2M.$$

Lower Bound Ingredient #2: Segmentation

⋮
operation
load
store
operation
load
load
operation
store
store
load
operation
store
load
⋮

segment i
iterations S_i
$|S_i| = 2$

For any segment i,

$$\begin{aligned} \mathcal{C}_i &\geq \mu(S_i) - 2M \\ &\geq |S_i|^{2/3} - 2M. \end{aligned}$$

Summing over all segments,

$$\mathcal{C} \geq \sum_i |S_i|^{2/3} - 2M.$$

If $|S_i| \approx M^{3/2}$ for all segments i,

$$\mathcal{C} \succeq \frac{N^3}{M^{3/2}} \cdot M = \frac{N^3}{M^{1/2}},$$

the stated lower bound.

Generalizing the Matrix Multiplication Loop Nest

Matrix multiplication:

for $i = 1 : N$
 for $j = 1 : N$
  for $k = 1 : N$
   $\mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}$

Parameter	**Description**	**Example: Matrix Mult.**
d	depth of loop nest	3

Generalizing the Matrix Multiplication Loop Nest

Matrix multiplication:

for $i = 1 : N$
  for $j = 1 : N$
    for $k = 1 : N$
      $\mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}$

Parameter	Description	Example: Matrix Mult.
d	depth of loop nest	3
Z	iteration space (subset of $\mathbb{Z}^d$)	$\{1, \ldots, N\}^3$

Generalizing the Matrix Multiplication Loop Nest

Matrix multiplication:

for $i = 1 : N$
  for $j = 1 : N$
    for $k = 1 : N$
      $\mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}$

Parameter	Description	Example: Matrix Mult.
d	depth of loop nest	3
Z	iteration space (subset of $\mathbb{Z}^d$)	$\{1, \ldots, N\}^3$
m	number of arrays	3

Generalizing the Matrix Multiplication Loop Nest

Matrix multiplication:

$$
\begin{array}{l}
\text{for } i = 1 : N \\
\quad \text{for } j = 1 : N \\
\quad\quad \text{for } k = 1 : N \\
\quad\quad\quad \mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}
\end{array}
$$

Parameter	**Description**	**Example: Matrix Mult.**
d	depth of loop nest	3
Z	iteration space (subset of $\mathbb{Z}^d$)	$\{1, \ldots, N\}^3$
m	number of arrays	3
$A_1, \ldots, A_m$	arrays	$\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$

Generalizing the Matrix Multiplication Loop Nest

Matrix multiplication:

$$
\begin{array}{l}
\text{for } i = 1 : N \\
\quad \text{for } j = 1 : N \\
\quad\quad \text{for } k = 1 : N \\
\quad\quad\quad \mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}
\end{array}
$$

Parameter	**Description**	**Example: Matrix Mult.**
d	depth of loop nest	3
Z	iteration space (subset of $\mathbb{Z}^d$)	$\{1, \ldots, N\}^3$
m	number of arrays	3
$A_1, \ldots, A_m$	arrays	$\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$
$d_1, \ldots, d_m$	array dimensions	2, 2, 2

Generalizing the Matrix Multiplication Loop Nest

Matrix multiplication:

for $i = 1 : N$
  for $j = 1 : N$
    for $k = 1 : N$
      $\mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}$

Parameter	**Description**	**Example: Matrix Mult.**
d	depth of loop nest	3
Z	iteration space (subset of $\mathbb{Z}^d$)	$\{1, \ldots, N\}^3$
m	number of arrays	3
$A_1, \ldots, A_m$	arrays	$\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$
$d_1, \ldots, d_m$	array dimensions	2, 2, 2
$\phi_1, \ldots, \phi_m$	array subscripts (linear maps: $\mathbb{Z}^d \to \mathbb{Z}^{d_j}$)	$(i, j, k) \mapsto (i, k)$, (k, j), (i, j)

Generalizing the Matrix Multiplication Loop Nest

Matrix multiplication:

$$
\begin{array}{l}
\text{for } i = 1 : N \\
\quad \text{for } j = 1 : N \\
\quad\quad \text{for } k = 1 : N \\
\quad\quad\quad \mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}
\end{array}
$$

Parameter	**Description**	**Example: Matrix Mult.**
d	depth of loop nest	3
Z	iteration space (subset of $\mathbb{Z}^d$)	$\{1, \ldots, N\}^3$
m	number of arrays	3
$A_1, \ldots, A_m$	arrays	**A**, **B**, **C**
$d_1, \ldots, d_m$	array dimensions	2, 2, 2
$\phi_1, \ldots, \phi_m$	array subscripts (linear maps: $\mathbb{Z}^d \to \mathbb{Z}^{d_j}$)	$(i, j, k) \mapsto (i, k), (k, j), (i, j)$

General loop nest:

$$
\begin{array}{l}
\text{for } i = (i_1, \ldots, i_d) \in Z \\
\quad \text{inner\_loop}_i(A_1(\phi_1(i)), \ldots, A_m(\phi_m(i)))
\end{array}
$$

General Loop Nests: Main Result

General loop nest:

$$\begin{array}{l} \text{for } i = (i_1, \ldots, i_d) \in Z \\ \quad \text{inner\_loop}_i(A_1(\phi_1(i)), \ldots, A_m(\phi_m(i))) \end{array}$$

Main Result: Communication Lower Bound for General Loop Nests

LB: There exists $\alpha \geq 0$ such that, in any implementation, $\mathcal{C} \succeq \frac{|Z|}{M^\alpha}$.

UB: There exists a tiled implementation such that $\mathcal{C} \preceq \frac{|Z|}{M^\alpha}$.

General Loop Nests: Main Result

General loop nest:

for $i = (i_1, \ldots, i_d) \in Z$
  inner_loop$_i(A_1(\phi_1(i)), \ldots, A_m(\phi_m(i)))$

Main Result: Communication Lower Bound for General Loop Nests

LB: There exists $\alpha \geq 0$ such that, in any implementation, $\mathcal{C} \succeq \frac{|Z|}{M^\alpha}$.

UB: There exists a tiled implementation such that $\mathcal{C} \preceq \frac{|Z|}{M^\alpha}$.

Exponent α and tile params. are computable from the subscripts $(\phi_j)_j$.

- Exponent α is a rational number between 0 and $m-1$.
- Tiles are parallelotopes (in $\mathbb{Z}^d$) of size $\approx M^{\alpha+1}$.
- Matrix multiply: $\alpha = 1/2$; tiles are cubes of size $\approx M^{3/2}$.

New Ingredient: Hölder-Brascamp-Lieb (HBL) Inequality

Continuing notations $d, m, \phi_1, \ldots, \phi_m$ from the general loop nest,

Hölder-Brascamp-Lieb (HBL) Inequality

If there exists $s = (s_1, \ldots, s_m) \in [0,1]^m$ such that,

for all subgroups H of $\mathbb{Z}^d$, $$\operatorname{rank} H \leq \sum_{j=1}^{m} s_j \cdot \operatorname{rank} \phi_j(H),$$

then,

for all subsets S of $\mathbb{Z}^d$, $$|S| \leq \prod_{j=1}^{m} |\phi_j(S)|^{s_j}.$$

New Ingredient: Hölder-Brascamp-Lieb (HBL) Inequality

Continuing notations $d, m, \phi_1, \ldots, \phi_m$ from the general loop nest,

Hölder-Brascamp-Lieb (HBL) Inequality

If there exists $s = (s_1, \ldots, s_m) \in [0,1]^m$ such that,

for all subgroups H of $\mathbb{Z}^d$, $\qquad \operatorname{rank} H \leq \sum_{j=1}^{m} s_j \cdot \operatorname{rank} \phi_j(H)$,

then,

for all subsets S of $\mathbb{Z}^d$, $\qquad |S| \leq \prod_{j=1}^{m} |\phi_j(S)|^{s_j}$.

Loomis-Whitney inequality is a special case:

- $m = d = 3$ and $(\phi_1, \phi_2, \phi_3) = (\phi_{\mathbf{A}}, \phi_{\mathbf{B}}, \phi_{\mathbf{C}})$
- Hypothesis satisfied by $(s_1, s_2, s_3) = (1/2, 1/2, 1/2)$.

HBL-LP: A Linear Program for the Optimal Exponent

- HBL lets us bound below the memory footprint of iterations S,

$$\mu(S) = \sum_{j=1}^{m} |\phi_j(S)| \geq |S|^{1/\sigma(s)}, \quad \text{abbreviating} \;\; \sigma(s) = \sum_{j=1}^{m} s_j.$$

HBL-LP: A Linear Program for the Optimal Exponent

- HBL lets us bound below the memory footprint of iterations S,

$$\mu(S) = \sum_{j=1}^{m} |\phi_j(S)| \geq |S|^{1/\sigma(s)}, \quad \text{abbreviating } \sigma(s) = \sum_{j=1}^{m} s_j.$$

- The segmentation argument yields a communication lower bound,

$$\mathcal{C} \succeq \frac{|Z|}{M^{\sigma(s)-1}}, \quad \text{meaning that the exponent } \alpha \leq \sigma(s) - 1.$$

HBL-LP: A Linear Program for the Optimal Exponent

- HBL lets us bound below the memory footprint of iterations S,

$$\mu(S) = \sum_{j=1}^{m} |\phi_j(S)| \geq |S|^{1/\sigma(s)}, \quad \text{abbreviating } \sigma(s) = \sum_{j=1}^{m} s_j.$$

- The segmentation argument yields a communication lower bound,

$$\mathcal{C} \succeq \frac{|Z|}{M^{\sigma(s)-1}}, \quad \text{meaning that the exponent } \alpha \leq \sigma(s) - 1.$$

- A lower bound of this form exists for any $s \in \mathcal{P}$, where

$$\mathcal{P} = \left\{ s \in [0,1]^m \;\middle|\; (\forall H \leq \mathbb{Z}^d) \ \operatorname{rank} H \leq \sum_{j=1}^{m} s_j \cdot \operatorname{rank} \phi_j(H) \right\};$$

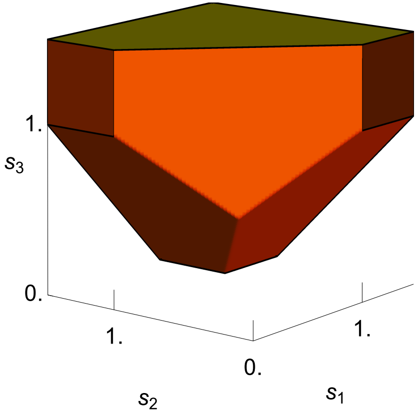
  to find the sharpest such bound, we with to solve a linear program,

$$\alpha + 1 = \min \left\{ \sigma(s) \;\middle|\; s \in \mathcal{P} \right\}.$$

HBL-LP: Challenges and a Solution Algorithm

(Bad news) Checking membership in $\mathcal{P}$ nominally requires considering all (infinitely many) subgroups $H \leq \mathbb{Z}^d$.

$\mathcal{P}$ for matrix multiplication:

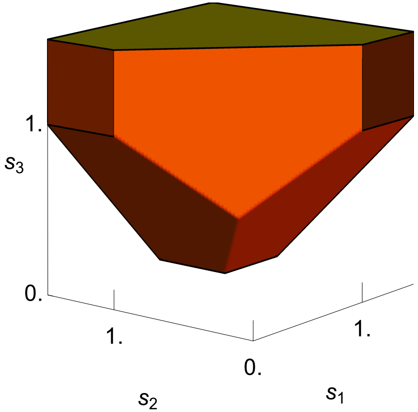


HBL-LP: Challenges and a Solution Algorithm

(Bad news) Checking membership in $\mathcal{P}$ nominally requires considering all (infinitely many) subgroups $H \leq \mathbb{Z}^d$.

(Good news) The subgroup ranks are integers between 0 and d: the linear constraints defining $\mathcal{P}$ are elements of a finite set.

$\mathcal{P}$ for matrix multiplication:



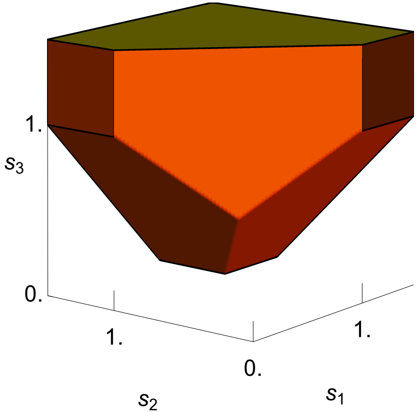
HBL-LP: Challenges and a Solution Algorithm

(Bad news) Checking membership in $\mathcal{P}$ nominally requires considering all (infinitely many) subgroups $H \leq \mathbb{Z}^d$.

(Good news) The subgroup ranks are integers between 0 and d: the linear constraints defining $\mathcal{P}$ are elements of a finite set.

(Bad news?) The existence of an algorithm that decides whether a particular constraint is involved in the HBL-LP is equivalent to Hilbert's Tenth Problem over $\mathbb{Q}$, conjectured to have a negative answer.

$\mathcal{P}$ for matrix multiplication:



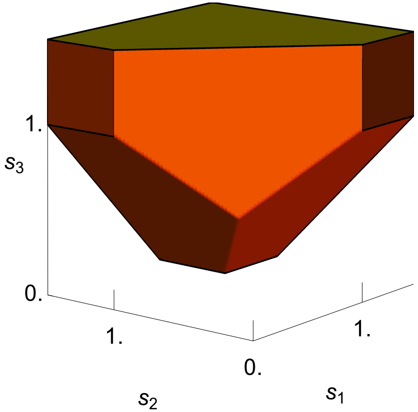
HBL-LP: Challenges and a Solution Algorithm

(Bad news) Checking membership in $\mathcal{P}$ nominally requires considering all (infinitely many) subgroups $H \leq \mathbb{Z}^d$.

(Good news) The subgroup ranks are integers between 0 and d: the linear constraints defining $\mathcal{P}$ are elements of a finite set.

(Bad news?) The existence of an algorithm that decides whether a particular constraint is involved in the HBL-LP is equivalent to Hilbert's Tenth Problem over $\mathbb{Q}$, conjectured to have a negative answer.

$\mathcal{P}$ for matrix multiplication:



Good News

Membership in $\mathcal{P}$ is decidable.

Tiling with Parallelotopes

Existence of an Optimal Tiling

There exist parallelotopes T satisfying $|T| \succeq M^{\alpha+1}$ and $\mu(T) \leq M$.

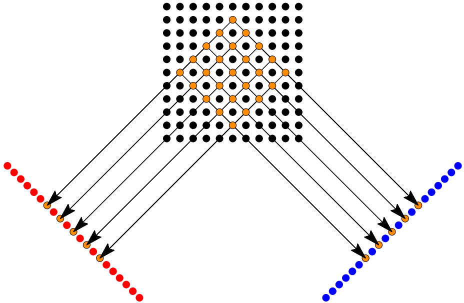
Ingredients: linear duality, subgroup series (flags), Smith normal form.

Tiling with Parallelotopes

Existence of an Optimal Tiling

There exist parallelotopes T satisfying $|T| \succeq M^{\alpha+1}$ and $\mu(T) \leq M$.

Ingredients: linear duality, subgroup series (flags), Smith normal form.



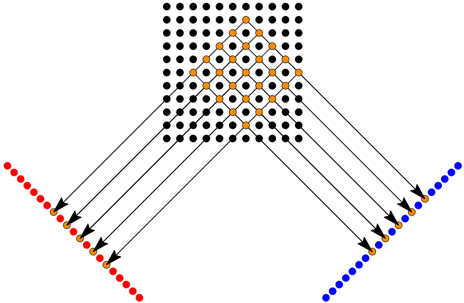
Access $\phi_1 = i - j$, $\phi_2 = i + j$

Tiling with Parallelotopes

Existence of an Optimal Tiling

There exist parallelotopes T satisfying $|T| \succeq M^{\alpha+1}$ and $\mu(T) \leq M$.

Ingredients: linear duality, subgroup series (flags), Smith normal form.



Access $\phi_1 = i - j$, $\phi_2 = i + j$

Example: Matrix Multiplication (Revisited)

Loop nest:

for $i = 1 : N$
  for $j = 1 : N$
    for $k = 1 : N$
      $\mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}$

Parameters:

$$d = 3, \quad Z = \{1, \dots, N\}^3, \quad m = 3,$$

$$\begin{array}{ccccc}
(i,j,k) & \xmapsto{\phi_1} & (i,k) & \xmapsto{A_1} & \mathbf{A}_{i,k} \\
(i,j,k) & \xmapsto{\phi_2} & (k,j) & \xmapsto{A_2} & \mathbf{B}_{k,j} \\
(i,j,k) & \xmapsto{\phi_3} & (i,j) & \xmapsto{A_3} & \mathbf{C}_{i,j}
\end{array}$$

Example: Matrix Multiplication (Revisited)

Loop nest:

for $i = 1 : N$
  for $j = 1 : N$
    for $k = 1 : N$
      $\mathbf{C}_{i,j} \leftarrow \mathbf{C}_{i,j} + \mathbf{A}_{i,k} * \mathbf{B}_{k,j}$

Parameters:

$$d = 3, \quad Z = \{1, \ldots, N\}^3, \quad m = 3,$$

$$\begin{array}{ccccc} (i,j,k) & \xmapsto{\phi_1} & (i,k) & \xmapsto{A_1} & \mathbf{A}_{i,k} \\ (i,j,k) & \xmapsto{\phi_2} & (k,j) & \xmapsto{A_2} & \mathbf{B}_{k,j} \\ (i,j,k) & \xmapsto{\phi_3} & (i,j) & \xmapsto{A_3} & \mathbf{C}_{i,j} \end{array}$$

HBL theory gives:

- Exponent $\alpha = 1/2$ (data reuse $\preceq M^{1/2}$);
- Tile $T = \{1, \ldots, b\}^3$, with $b \approx M^{1/2}$.

Communication Lower Bound

The lower bound $\mathcal{C} \succeq \dfrac{N^3}{M^{1/2}}$ is attainable by tiling with translates of T.

Example: Matrix-Vector Multiplication

Loop nest:

for $i = 1 : N$
  for $j = 1 : N$
    $\mathbf{y}_i \leftarrow \mathbf{y}_i + \mathbf{A}_{i,j} * \mathbf{x}_j$

Parameters:

$$d = 2, \quad Z = \{1, \ldots, N\}^2, \quad m = 3,$$

$$\begin{array}{ccccc} (i,j) & \xmapsto{\phi_1} & (i,j) & \xmapsto{A_1} & \mathbf{A}_{i,j} \\ (i,j) & \xmapsto{\phi_2} & (j) & \xmapsto{A_2} & \mathbf{x}_j \\ (i,j) & \xmapsto{\phi_3} & (i) & \xmapsto{A_3} & \mathbf{y}_i \end{array}$$

Example: Matrix-Vector Multiplication

Loop nest:

$$\begin{array}{l} \text{for } i = 1 : N \\ \quad \text{for } j = 1 : N \\ \quad\quad \mathbf{y}_i \leftarrow \mathbf{y}_i + \mathbf{A}_{i,j} * \mathbf{x}_j \end{array}$$

Parameters:

$$d = 2, \quad Z = \{1, \ldots, N\}^2, \quad m = 3,$$

$$\begin{array}{ccccc} (i,j) & \xmapsto{\phi_1} & (i,j) & \xmapsto{A_1} & \mathbf{A}_{i,j} \\ (i,j) & \xmapsto{\phi_2} & (j) & \xmapsto{A_2} & \mathbf{x}_j \\ (i,j) & \xmapsto{\phi_3} & (i) & \xmapsto{A_3} & \mathbf{y}_i \end{array}$$

HBL theory gives:

- Exponent $\alpha = 0$ (data reuse $\preceq 1$);
- Tile $T = \{(1,1,1)\}^3$.

Communication Lower Bound

The lower bound $\mathcal{C} \succeq N^2$ is attainable by tiling with translates of T.

Example: Tensor Contraction

Loop nest:

for $i = 1 : N$
  for $j = 1 : N$
    for $k = 1 : N$
      for $l = 1 : N$
        for $m = 1 : N$
          $\mathbf{C}_{i,j,k} \leftarrow \mathbf{C}_{i,j,k} + \mathbf{A}_{i,l,m} * \mathbf{B}_{l,m,j,k}$

Parameters:

$d = 5, \quad Z = \{1, \ldots, N\}^5, \quad m = 3,$

$$\begin{array}{ccccc} (i,j,k,l,m) & \xmapsto{\phi_1} & (i,l,m) & \xmapsto{A_1} & \mathbf{A}_{i,l,m} \\ (i,j,k,l,m) & \xmapsto{\phi_2} & (l,m,j,k) & \xmapsto{A_2} & \mathbf{B}_{l,m,j,k} \\ (i,j,k,l,m) & \xmapsto{\phi_3} & (i,j,k) & \xmapsto{A_3} & \mathbf{C}_{i,j,k} \end{array}$$

Example: Tensor Contraction

Loop nest:

for $i = 1 : N$
  for $j = 1 : N$
    for $k = 1 : N$
      for $l = 1 : N$
        for $m = 1 : N$
          $\mathbf{C}_{i,j,k} \leftarrow \mathbf{C}_{i,j,k} + \mathbf{A}_{i,l,m} * \mathbf{B}_{l,m,j,k}$

Parameters:

$$d = 5, \quad Z = \{1, \ldots, N\}^5, \quad m = 3,$$

$$\begin{array}{ccccc} (i,j,k,l,m) & \overset{\phi_1}{\longmapsto} & (i,l,m) & \overset{A_1}{\longmapsto} & \mathbf{A}_{i,l,m} \\ (i,j,k,l,m) & \overset{\phi_2}{\longmapsto} & (l,m,j,k) & \overset{A_2}{\longmapsto} & \mathbf{B}_{l,m,j,k} \\ (i,j,k,l,m) & \overset{\phi_3}{\longmapsto} & (i,j,k) & \overset{A_3}{\longmapsto} & \mathbf{C}_{i,j,k} \end{array}$$

HBL theory gives:

- Exponent $\alpha = 1/2$ (data reuse $\preceq M^{1/2}$, same as matrix multiply!);
- Tile $T = \bigtimes_{i=1}^{5} \{1, \ldots, b_i\}$ with $b_1,\ b_2 b_3,\ b_4 b_5 \approx M^{1/2}$.

Communication Lower Bound

The lower bound $\mathcal{C} \succeq \dfrac{N^5}{M^{1/2}}$ is attainable by tiling with translates of T.

Example: Particle Simulation

Loop nest:

for $i = 1 : N$
  for $j = 1 : N$
    $\mathbf{V}_i \leftarrow \mathbf{V}_i + G(\mathbf{P}_i, \mathbf{P}_j)$

Parameters:

$$d = 2, \quad Z = \{1, \ldots, N\}^2, \quad m = 3,$$

$$\begin{array}{ccccc} (i,j) & \xmapsto{\phi_1} & (i) & \xmapsto{A_1} & \mathbf{P}_i \\ (i,j) & \xmapsto{\phi_2} & (j) & \xmapsto{A_2} & \mathbf{P}_j \\ (i,j) & \xmapsto{\phi_3} & (i) & \xmapsto{A_3} & \mathbf{V}_i \end{array}$$

Example: Particle Simulation

Loop nest:

$$\begin{array}{l} \text{for } i = 1 : N \\ \quad \text{for } j = 1 : N \\ \qquad \mathbf{V}_i \leftarrow \mathbf{V}_i + G(\mathbf{P}_i, \mathbf{P}_j) \end{array}$$

Parameters:

$$d = 2, \quad Z = \{1, \ldots, N\}^2, \quad m = 3,$$

$$\begin{array}{ccccc} (i,j) & \xmapsto{\phi_1} & (i) & \xmapsto{A_1} & \mathbf{P}_i \\ (i,j) & \xmapsto{\phi_2} & (j) & \xmapsto{A_2} & \mathbf{P}_j \\ (i,j) & \xmapsto{\phi_3} & (i) & \xmapsto{A_3} & \mathbf{V}_i \end{array}$$

HBL theory gives:

- Exponent $\alpha = 1$ (data reuse $\preceq M$);
- Tile $T = \{1, \ldots, b\}^2$, with $b \approx M$.

Communication Lower Bound

The lower bound $\mathcal{C} \succeq \dfrac{N^2}{M}$ is attainable by tiling with translates of T.

Example: Convolutional Neural Network

Loop nest:

for $i = 1 : N$
  for $j = 1 : N$
    for $k = 1 : N$
      for $l = 1 : N$
        for $m = 1 : N$
          for $n = 1 : N$
            $\mathbf{C}_{i,j,m,n} \leftarrow \mathbf{C}_{i,j,m,n} + \mathbf{A}_{i,k,l} * \mathbf{B}_{j,k+m,l+n}$

Parameters:

$d = 6, \quad Z = \{1, \ldots, N\}^6, \quad m = 3,$

$$
\begin{array}{ccccc}
(i,j,k,l,m,n) & \xmapsto{\phi_1} & (i,k,l) & \xmapsto{A_1} & \mathbf{A}_{i,k,l} \\
(i,j,k,l,m,n) & \xmapsto{\phi_2} & (j,k+m,l+n) & \xmapsto{A_2} & \mathbf{B}_{j,k+m,l+n} \\
(i,j,k,l,m,n) & \xmapsto{\phi_3} & (i,j,m,n) & \xmapsto{A_3} & \mathbf{C}_{i,j,m,n}
\end{array}
$$

Example: Convolutional Neural Network

Loop nest:

```
for i = 1 : N
  for j = 1 : N
    for k = 1 : N
      for l = 1 : N
        for m = 1 : N
          for n = 1 : N
            C_{i,j,m,n} ← C_{i,j,m,n} + A_{i,k,l} * B_{j,k+m,l+n}
```

Parameters:

$d = 6, \quad Z = \{1, \ldots, N\}^6, \quad m = 3,$

$$\begin{array}{ccccc} (i,j,k,l,m,n) & \xmapsto{\phi_1} & (i,k,l) & \xmapsto{A_1} & \mathbf{A}_{i,k,l} \\ (i,j,k,l,m,n) & \xmapsto{\phi_2} & (j,k+m,l+n) & \xmapsto{A_2} & \mathbf{B}_{j,k+m,l+n} \\ (i,j,k,l,m,n) & \xmapsto{\phi_3} & (i,j,m,n) & \xmapsto{A_3} & \mathbf{C}_{i,j,m,n} \end{array}$$

HBL theory gives:

- Exponent $\alpha = 1$ (data reuse $\preceq M$, same as particle simulation);
- Tile $T = \times_{i=1}^{6}\{1, \ldots, b_i\}$ with $b_1, b_2 \approx 1,\ b_3, b_4, b_5, b_6 \approx M^{1/2}$.

Communication Lower Bound

The lower bound $\mathcal{C} \succeq \dfrac{N^6}{M}$ is attainable by tiling with translates of T.

Concluding Remarks

- Communication lower bounds for loop nests

Concluding Remarks

- Communication lower bounds for loop nests
- Lower bounds attainable by iteration space tiling

Concluding Remarks

- Communication lower bounds for loop nests
- Lower bounds attainable by iteration space tiling
- Goal: compiler automatically generates optimal code

Concluding Remarks

- Communication lower bounds for loop nests
- Lower bounds attainable by iteration space tiling
- Goal: compiler automatically generates optimal code

Many details omitted:

- Parallel implementations: replace $|Z|$ by $|Z|/P$

Concluding Remarks

- Communication lower bounds for loop nests
- Lower bounds attainable by iteration space tiling
- Goal: compiler automatically generates optimal code

Many details omitted:

- Parallel implementations: replace $|Z|$ by $|Z|/P$
- Constants hidden in asymptotic notation ($\preceq, \approx, \succeq$)

Concluding Remarks

- Communication lower bounds for loop nests
- Lower bounds attainable by iteration space tiling
- Goal: compiler automatically generates optimal code

Many details omitted:

- Parallel implementations: replace $|Z|$ by $|Z|/P$
- Constants hidden in asymptotic notation $(\preceq, \approx, \succeq)$
- Attainability when Z is irregular (e.g., sparse)

Concluding Remarks

- Communication lower bounds for loop nests
- Lower bounds attainable by iteration space tiling
- Goal: compiler automatically generates optimal code

Many details omitted:

- Parallel implementations: replace $|Z|$ by $|Z|/P$
- Constants hidden in asymptotic notation $(\preceq, \approx, \succeq)$
- Attainability when Z is irregular (e.g., sparse)
- Attainability when data dependences

Concluding Remarks

- Communication lower bounds for loop nests
- Lower bounds attainable by iteration space tiling
- Goal: compiler automatically generates optimal code

Many details omitted:

- Parallel implementations: replace $|Z|$ by $|Z|/P$
- Constants hidden in asymptotic notation $(\preceq, \approx, \succeq)$
- Attainability when Z is irregular (e.g., sparse)
- Attainability when data dependences
- Complexity of computing lower bound and optimal tiling

Concluding Remarks

- Communication lower bounds for loop nests
- Lower bounds attainable by iteration space tiling
- Goal: compiler automatically generates optimal code

Many details omitted:

- Parallel implementations: replace $|Z|$ by $|Z|/P$
- Constants hidden in asymptotic notation ($\preceq, \approx, \succeq$)
- Attainability when Z is irregular (e.g., sparse)
- Attainability when data dependences
- Complexity of computing lower bound and optimal tiling
- Optimizing subprograms, compositions of programs

Concluding Remarks

- Communication lower bounds for loop nests
- Lower bounds attainable by iteration space tiling
- Goal: compiler automatically generates optimal code

Many details omitted:

- Parallel implementations: replace $|Z|$ by $|Z|/P$
- Constants hidden in asymptotic notation $(\preceq, \approx, \succeq)$
- Attainability when Z is irregular (e.g., sparse)
- Attainability when data dependences
- Complexity of computing lower bound and optimal tiling
- Optimizing subprograms, compositions of programs

Please see tech. reports UCB/EECS-2013-61 and UCB/EECS-2015-185.

Thank You!