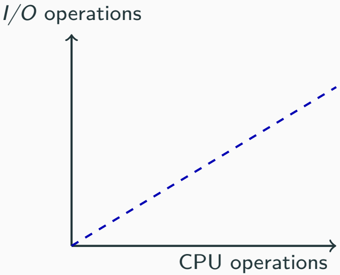
Automated Derivation of Parametric Data Movement Lower Bounds for Affine Programs

Auguste Olivry Julien Langou Louis-Noël Pouchet
P. Sadayappan Fabrice Rastello

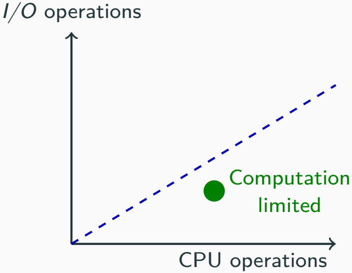
June 19, 2020

What does it mean to compute a data movement lower bound, and how is it different from optimizing a program's I/O?

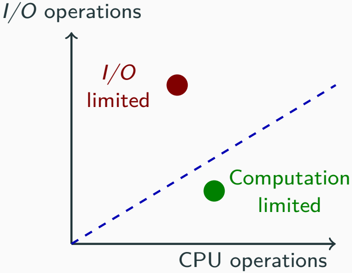
Operations vs. *I/O*



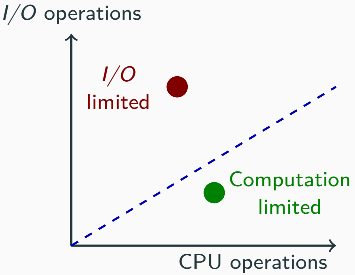
Operations vs. I/O



Operations vs. *I/O*

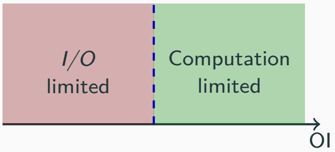


Operations vs. *I/O*



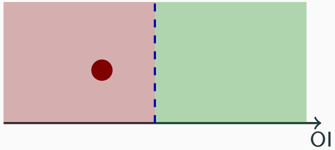
Operational intensity

$$\text{OI} = \frac{\#\ \text{CPU operations}}{\#\ I/O\ \text{operations}}$$



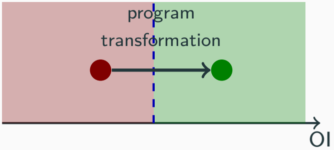
Optimizing compiler vs. IOLB

Optimizing compiler:
Improves a program's OI through program transformations (operation schedule and memory allocation)



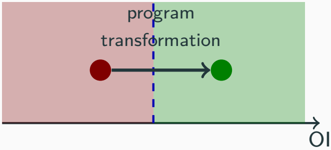
Optimizing compiler vs. IOLB

Optimizing compiler:
Improves a program's OI through program transformations (operation schedule and memory allocation)

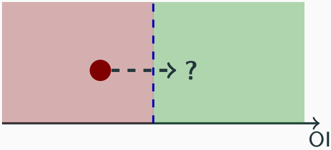


Optimizing compiler vs. IOLB

Optimizing compiler:
Improves a program's OI through program transformations (operation schedule and memory allocation)

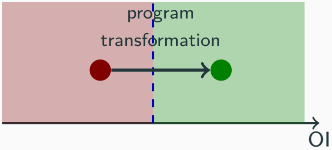


IOLB:
Proves that a program's OI cannot be improved beyond a certain limit

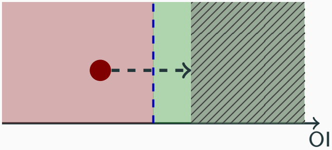


Optimizing compiler vs. IOLB

Optimizing compiler:
Improves a program's OI through program transformations (operation schedule and memory allocation)

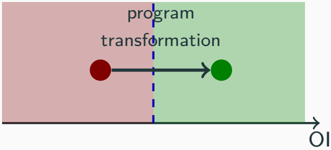


IOLB:
Proves that a program's OI cannot be improved beyond a certain limit

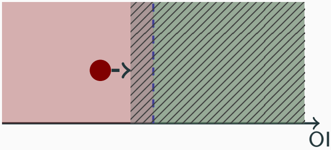


Optimizing compiler vs. IOLB

Optimizing compiler:
Improves a program's OI through program transformations (operation schedule and memory allocation)



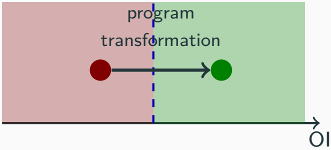
IOLB:
Proves that a program's OI cannot be improved beyond a certain limit

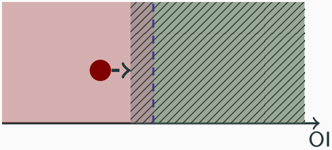


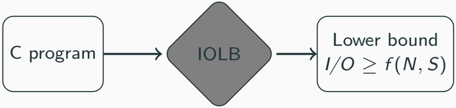
Optimizing compiler vs. IOLB

Optimizing compiler:
Improves a program's OI through program transformations (operation schedule and memory allocation)

IOLB:
Proves that a program's OI cannot be improved beyond a certain limit



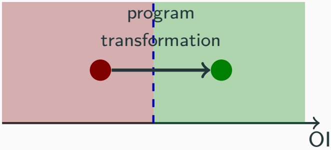


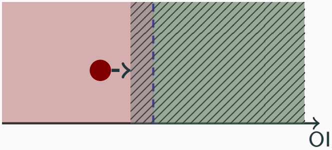


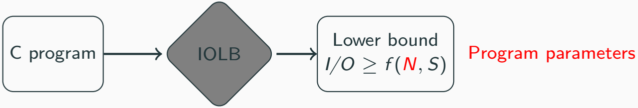
Optimizing compiler vs. IOLB

Optimizing compiler:
Improves a program's OI through program transformations (operation schedule and memory allocation)

IOLB:
Proves that a program's OI cannot be improved beyond a certain limit



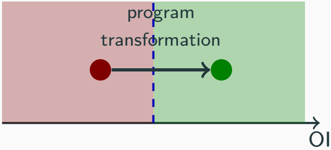


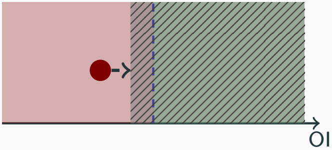


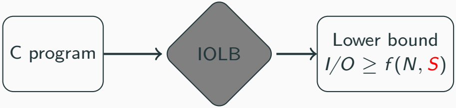
Optimizing compiler vs. IOLB

Optimizing compiler:
Improves a program's OI through program transformations (operation schedule and memory allocation)

IOLB:
Proves that a program's OI cannot be improved beyond a certain limit





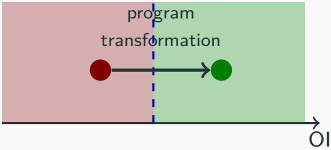


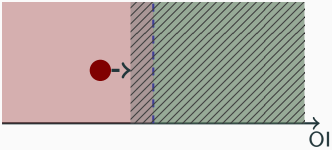
Cache size

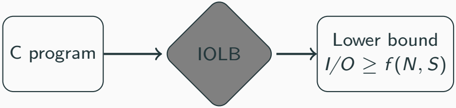
Optimizing compiler vs. IOLB

Optimizing compiler:
Improves a program's OI through program transformations (operation schedule and memory allocation)

IOLB:
Proves that a program's OI cannot be improved beyond a certain limit







I/O lower bound $\Leftrightarrow$ OI upper bound

Outline

I/O Complexity Lower Bound Proofs

- Model
- S-Partitioning
- Geometric Reasoning

Towards Automation

Overview of Our Work

Results

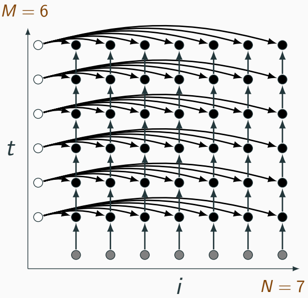
I/O Complexity Lower Bound Proofs

Model

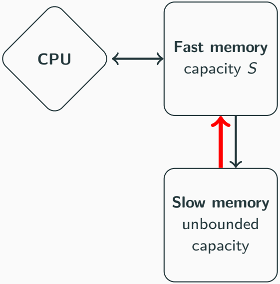
Computation model

- Computational directed acyclic graph (CDAG)
    - Program = elementary computations (vertices) with data dependencies (edges)
    - 1 computation = 1 piece of data
    - Sequential execution
    - Multiple valid *schedules*

```
Parameters: N, M;
Input: A[N], C[M]; Output: A[N];
for(t=0; t<M; t++)
  for (i=0; i<N; i++)
    A[i] = A[i] * C[t];
```



Model



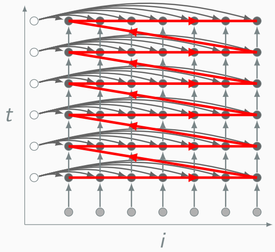
Cost model

- 2-level memory hierarchy
  - cost: #transfers from slow to fast memory (LOADS)
  - "manual" memory management

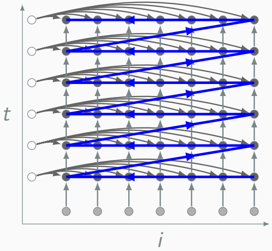
➡ **Data movement (*I/O*) complexity $Q =$ minimum cost over all valid schedules**

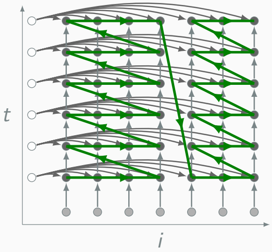
Schedules

```
Parameters: N, M;
Input: A[N], C[M]; Output: A[N];
for(t=0; t<M; t++)
  for (i=0; i<N; i++)
    A[i] = A[i] * C[t];
```



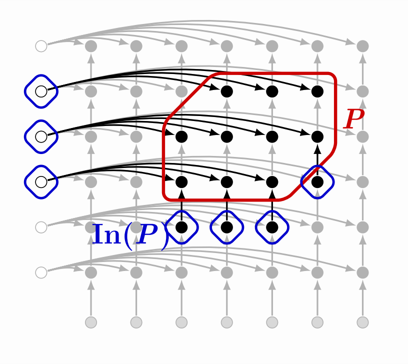
Original schedule





Any **schedule satisfying data dependencies is possible**

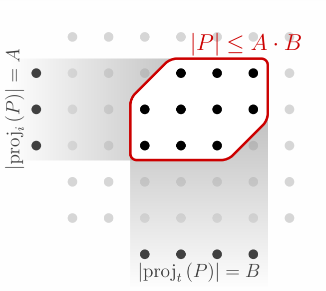
S-partitioning




```
Parameters: N, M;
Input: A[N], C[M]; Output: A[N];
for(t=0; t<M; t++)
  for (i=0; i<N; i++)
    A[i] = A[i] * C[t];
```

- In-set of a subgraph P: set of predecessors of P outside P
- Upper bound on the size of P with $\mathrm{In}(P) \leq 2S \Leftrightarrow I/O$ lower bound

Geometric reasoning



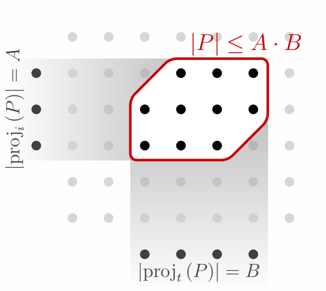

```
Parameters: N, M;
Input: A[N], C[M]; Output: A[N];
for(t=0; t<M; t++)
  for (i=0; i<N; i++)
    A[i] = A[i] * C[t];
```

- Natural embedding in a geometric space
- Regular dependencies: In-set size bounded by projection cardinality

Each projection of P is of cardinality $\leq 2S \Rightarrow |P| \leq (2S)^2$.

$$\Rightarrow Q \geq \lfloor NM/4S^2 \rfloor \cdot S \approx \frac{NM}{4S}.$$

Geometric reasoning




```
Parameters: N, M;
Input: A[N], C[M]; Output: A[N];
for(t=0; t<M; t++)
  for (i=0; i<N; i++)
    A[i] = A[i] * C[t];
```

- Natural embedding in a geometric space
- Regular dependencies: In-set size bounded by projection cardinality

Each projection of P is of cardinality $\leq 2S \Rightarrow |P| \leq (2S)^2$.

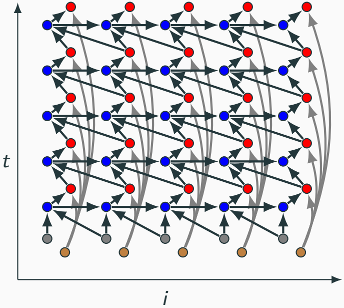
$$\Rightarrow Q \geq \lfloor NM/4S^2 \rfloor \cdot S \approx \frac{NM}{4S}.$$

➡ Generalizes to arbitrary dimensions (Discrete Brascamp-Lieb inequality [2])

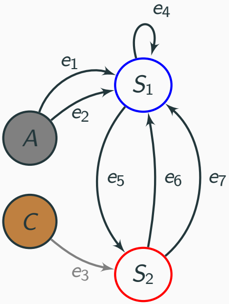
Towards Automation

Compact representation for affine programs

- Data-flow graph (DFG): compact parametric representation of the CDAG for *affine* programs

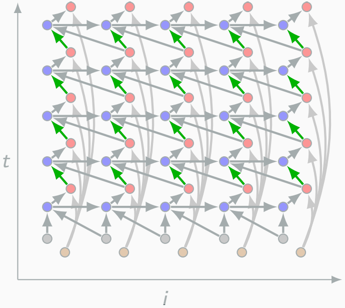



```
    for(t=0; t<T; t++)
      for (i=1; i<N-1; i++) {
S1:       A[i] = A[i-1] + A[i] + A[i+1];
S2:       A[i] *= C[i];
      }
```

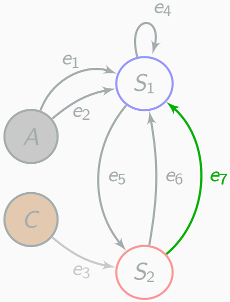


Compact representation for affine programs

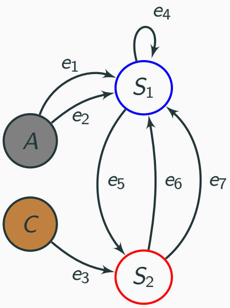
- Data-flow graph (DFG): compact parametric representation of the CDAG for *affine* programs




```
    for(t=0; t<T; t++)
      for (i=1; i<N-1; i++) {
S1:        A[i] = A[i-1] + A[i] + A[i+1];
S2:        A[i] *= C[i];
      }
```



DFG and Polyhedral Compilation Tools



$$D_A = [N] \rightarrow \{A[i] : \ 1 \leq i < N-1\}$$
$$D_C = [N] \rightarrow \{C[i] : \ 1 \leq i < N-1\}$$
$$D_{S_1} = [T, N] \rightarrow \{S_1[t, i] : \ 0 \leq t < T \ \wedge \ 1 \leq i < N-1\}$$
$$\left| D_{S_1} \right| = T(N-2)$$

Node domains

$$R_{e_1} = [N] \rightarrow \{A[i] \rightarrow S_1[0, i-1] : \ 2 \leq i < N-1\}$$
$$R_{e_2} = [N] \rightarrow \{A[i] \rightarrow S_1[0, i] : \ 1 \leq i < N-1\}$$

$$R_{e_3} = [T, N] \rightarrow \{C[i] \rightarrow S_2[t, i] : \ 0 \leq t < T \ \wedge \ 1 \leq i < N-1\}$$

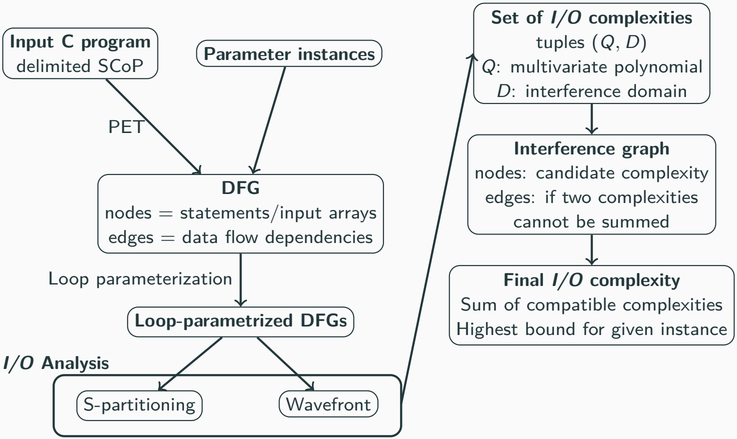
$$R_{e_4} = [T, N] \rightarrow \{S_1[t, i] \rightarrow S_1[t, i+1] : \ 0 \leq t < T \ \wedge \ 1 \leq i < N-2\}$$
$$R_{e_5} = [T, N] \rightarrow \{S_1[t, i] \rightarrow S_2[t, i] : \ 0 \leq t < T \ \wedge \ 1 \leq i < N-1\}$$
$$R_{e_6} = [T, N] \rightarrow \{S_2[t, i] \rightarrow S_1[t+1, i-1] : \ 0 \leq t < T-1 \ \wedge \ 2 \leq i < N-1\}$$
$$R_{e_7} = [T, N] \rightarrow \{S_2[t, i] \rightarrow S_1[t+1, i] : \ 0 \leq t < T-1 \ \wedge \ 1 \leq i < N-1\}$$

Edge relations

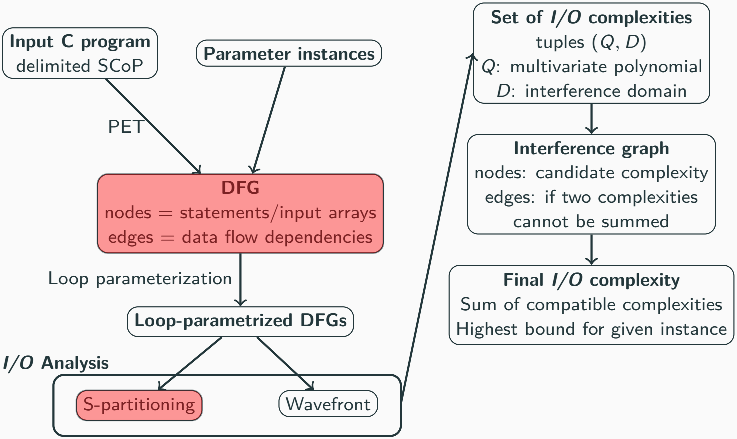
- ISL (Integer Set Library) representation
- computations = points in a multi-dimensional lattice
- memory dependencies = vectors
- Captures geometric regularity

Overview of Our Work

Overall framework



Overall framework



Results

Results on PolyBench/C

kernel	# input data	# ops	$Q^{\infty}_{\mathrm{low}}$	$OI_{\mathrm{manual}} \leq OI \leq OI_{\mathrm{up}}$	ratio	Published OI_{up}
2mm	$N_i N_k + N_k N_j + N_j N_l + N_i N_l$	$2(NiN_j N_k + N_i N_j N_l)$	$2(N_i N_j N_k + N_i N_j N_l)/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	–
3mm	$N_i N_k + N_k N_j + N_j N_m + N_m N_l$	$2(NiN_j N_k + N_j N_l N_m + N_i N_j N_l)$	$2(N_i N_j N_k + N_i N_j N_l + N_j N_l N_m)/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	–
cholesky	$\frac{1}{2} N^2$	$\frac{1}{3} N^3$	$\frac{1}{6} N^3/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	$8\sqrt{S}$ [1]
correlation	MN	$M^2 N$	$\frac{1}{2} M^2 N/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	–
covariance	MN	$M^2 N$	$\frac{1}{2} M^2 N/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	–
doitgen	$N_p N_q N_r + N_p^2$	$2N_p^2 N_q N_r$	$2N_p^2 N_q N_r/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	–
fdtd-2d	$3N_x N_y + T$	$11 N_x N_y T$	$\frac{2}{3\sqrt{3}} N_x N_y T/\sqrt{S}$	$\frac{11}{24}\sqrt{3}\sqrt{S} \leq OI \leq \frac{33}{2}\sqrt{3}\sqrt{S}$	36	–
floyd-warshall	N^2	$2N^3$	$2N^3/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
gemm	$N_i N_j + N_j N_k + N_i N_k$	$2N_i N_j N_k$	$2N_i N_j N_k/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$\sqrt{S}$ [4]
heat-3d	N^3	$30N^3 T$	$\frac{3}{8}\sqrt[3]{2} N^3 T/\sqrt[3]{S}$	$\frac{5}{2}\sqrt[3]{S} \leq OI \leq 40 \cdot 2^{2/3}\sqrt[3]{S}$	$16 \cdot 2^{2/3}$	–
jacobi-1d	N	$6NT$	$\frac{1}{4} NT/S$	$\frac{3}{2} S \leq OI \leq 24S$	16	$48S$ [3]
jacobi-2d	N^2	$10N^2 T$	$\frac{2}{3\sqrt{3}} N^2 T/\sqrt{S}$	$\frac{5}{4}\sqrt{S} \leq OI \leq 15\sqrt{3}\sqrt{S}$	$12\sqrt{3}$	$40\sqrt{2}\sqrt{S}$ [3]
lu	N^2	$\frac{2}{3} N^3$	$\frac{2}{3} N^3/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
ludcmp	N^2	$\frac{2}{3} N^3$	$\frac{2}{3} N^3/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
seidel-2d	N^2	$9N^2 T$	$\frac{2}{3\sqrt{3}} N^2 T/\sqrt{S}$	$\frac{9}{4}\sqrt{S} \leq OI \leq \frac{27\sqrt{3}}{2}\sqrt{S}$	$6\sqrt{3}$	–
symm	$\frac{1}{2} M^2 + 2MN$	$2M^2 N$	$2M^2 N/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
syr2k	$\frac{1}{2} N^2 + 2MN$	$2MN^2$	$MN^2/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	$8\sqrt{S}$ [1]
syrk	$\frac{1}{2} N^2 + MN$	MN^2	$\frac{1}{2} MN^2/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	$8\sqrt{S}$ [1]
trmm	$\frac{1}{2} M^2 + MN$	$M^2 N$	$M^2 N/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
atax	MN	$4MN$	MN	$4 \leq OI \leq 4$	1	–
bicg	MN	$4MN$	MN	$4 \leq OI \leq 4$	1	–
deriche	HW	$32HW$	HW	$\frac{16}{3} \leq OI \leq 32$	6	–
gemver	N^2	$10N^2$	N^2	$5 \leq OI \leq 10$	2	–
gesummv	$2N^2$	$4N^2$	$2N^2$	$2 \leq OI \leq 2$	1	–
mvt	N^2	$4N^2$	N^2	$4 \leq OI \leq 4$	1	–
trisolv	$\frac{1}{2} N^2$	N^2	$\frac{1}{2} N^2$	$2 \leq OI \leq 2$	1	–
adi	N^2	$30N^2 T$	$N^2 T$	$5 \leq OI \leq 30$	6	–
durbin	N	$2N^2$	$\frac{1}{2} N^2$	$\frac{2}{3} \leq OI \leq 4$	6	–
gramschmidt	MN	$2MN^2$	$MN^2/\sqrt{S}$	$1 \leq OI \leq 2\sqrt{S}$	$2\sqrt{S}$	–
nussinov	$\frac{1}{2} N^2$	$\frac{1}{3} N^3$	$\frac{1}{6} N^3/\sqrt{S}$	$1 \leq OI \leq 2\sqrt{S}$	$2\sqrt{S}$	–

Results on PolyBench/C

kernel	# input data	# ops	Q^{∞}_{low}	$OI_{\text{manual}} \leq OI \leq OI_{\text{up}}$	ratio	Published OI_{up}
2mm	$N_iN_k + N_kN_j + N_jN_l + N_iN_l$	$2(NiN_jN_k + N_iN_jN_l)$	$2(N_iN_jN_k + N_iN_jN_l)/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	–
3mm	$N_iN_k + N_kN_j + N_jN_m + N_mN_l$	$2(NiN_jN_k + N_jN_lN_m + N_iN_jN_l)$	$2(N_iN_jN_k + N_iN_jN_l + N_jN_lN_m)/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	–
cholesky	$\frac{1}{2}N^2$	$\frac{1}{3}N^3$	$\frac{1}{6}N^3/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	$8\sqrt{S}$ [1]
correlation	MN	M^2N	$\frac{1}{2}M^2N/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	–
covariance	MN	M^2N	$\frac{1}{2}M^2N/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	–
doitgen	$N_pN_qN_r + N_p^2$	$2N_p^2N_qN_r$	$2N_p^2N_qN_r/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	–
fdtd-2d	$3N_xN_y + T$	$11N_xN_yT$	$\frac{2}{3\sqrt{3}}N_xN_yT/\sqrt{S}$	$\frac{11}{24}\sqrt{3}\sqrt{S} \leq OI \leq \frac{33}{2}\sqrt{3}\sqrt{S}$	36	–
floyd-warshall	N^2	$2N^3$	$2N^3/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
gemm	$N_iN_j + N_jN_k + N_iN_k$	$2N_iN_jN_k$	$2N_iN_jN_k/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$\sqrt{S}$ [4]
heat-3d	N^3	$30N^3T$	$\frac{3}{8}\sqrt[3]{2}N^3T/\sqrt[3]{S}$	$\frac{5}{2}\sqrt[3]{S} \leq OI \leq 40 \cdot 2^{2/3}\sqrt[3]{S}$	$16 \cdot 2^{2/3}$	–
jacobi-1d	N	$6NT$	$\frac{1}{4}NT/S$	$\frac{3}{2}S \leq OI \leq 24S$	16	$48S$ [3]
jacobi-2d	N^2	$10N^2T$	$\frac{2}{3\sqrt{3}}N^2T/\sqrt{S}$	$\frac{5}{4}\sqrt{S} \leq OI \leq 15\sqrt{3}\sqrt{S}$	$12\sqrt{3}$	$40\sqrt{2}\sqrt{S}$ [3]
lu	N^2	$\frac{2}{3}N^3$	$\frac{2}{3}N^3/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
ludcmp	N^2	$\frac{2}{3}N^3$	$\frac{2}{3}N^3/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
seidel-2d	N^2	$9N^2T$	$\frac{2}{3\sqrt{3}}N^2T/\sqrt{S}$	$\frac{9}{4}\sqrt{S} \leq OI \leq \frac{27\sqrt{3}}{2}\sqrt{S}$	$6\sqrt{3}$	–
symm	$\frac{1}{2}M^2 + 2MN$	$2M^2N$	$2M^2N/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
syr2k	$\frac{1}{2}N^2 + 2MN$	$2MN^2$	$MN^2/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	$8\sqrt{S}$ [1]
syrk	$\frac{1}{2}N^2 + MN$	MN^2	$\frac{1}{2}MN^2/\sqrt{S}$	$\sqrt{S} \leq OI \leq 2\sqrt{S}$	2	$8\sqrt{S}$ [1]
trmm	$\frac{1}{2}M^2 + MN$	M^2N	$M^2N/\sqrt{S}$	$\sqrt{S} \leq OI \leq \sqrt{S}$	1	$8\sqrt{S}$ [1]
atax	MN	$4MN$	MN	$4 \leq OI \leq 4$	1	–
bicg	MN	$4MN$	MN	$4 \leq OI \leq 4$	1	–
deriche	HW	$32HW$	HW	$\frac{16}{3} \leq OI \leq 32$	6	–
gemver	N^2	$10N^2$	N^2	$5 \leq OI \leq 10$	2	–
gesummv	$2N^2$	$4N^2$	$2N^2$	$2 \leq OI \leq 2$	1	–
mvt	N^2	$4N^2$	N^2	$4 \leq OI \leq 4$	1	–
trisolv	$\frac{1}{2}N^2$	N^2	$\frac{1}{2}N^2$	$2 \leq OI \leq 2$	1	–
adi	N^2	$30N^2T$	N^2T	$5 \leq OI \leq 30$	6	–
durbin	N	$2N^2$	$\frac{1}{2}N^2$	$\frac{2}{3} \leq OI \leq 4$	6	–
gramschmidt	MN	$2MN^2$	$MN^2/\sqrt{S}$	$1 \leq OI \leq 2\sqrt{S}$	$2\sqrt{S}$	–
nussinov	$\frac{1}{2}N^2$	$\frac{1}{3}N^3$	$\frac{1}{6}N^3/\sqrt{S}$	$1 \leq OI \leq 2\sqrt{S}$	$2\sqrt{S}$	–

Improves or matches previously published bounds for all kernels

Results on PolyBench/C

kernel	# input data	# ops	Q^{∞}_{low}	$OI_{\text{manual}} \le OI \le OI_{\text{up}}$	ratio	Published OI_{up}
2mm	$N_iN_k + N_kN_j + N_jN_l + N_iN_l$	$2(NiN_jN_k + N_iN_jN_l)$	$2(N_iN_jN_k + N_iN_jN_l)/\sqrt{S}$	$\sqrt{S} \le OI \le \sqrt{S}$	1	–
3mm	$N_iN_k + N_kN_j + N_jN_m + N_mN_l$	$2(NiN_jN_k + N_jN_lN_m + N_iN_jN_l)$	$2(N_iN_jN_k + N_iN_jN_l + N_jN_lN_m)/\sqrt{S}$	$\sqrt{S} \le OI \le \sqrt{S}$	1	–
cholesky	$\frac{1}{2}N^2$	$\frac{1}{3}N^3$	$\frac{1}{6}N^3/\sqrt{S}$	$\sqrt{S} \le OI \le 2\sqrt{S}$	2	$8\sqrt{S}$ [1]
correlation	MN	M^2N	$\frac{1}{2}M^2N/\sqrt{S}$	$\sqrt{S} \le OI \le 2\sqrt{S}$	2	–
covariance	MN	M^2N	$\frac{1}{2}M^2N/\sqrt{S}$	$\sqrt{S} \le OI \le 2\sqrt{S}$	2	–
doitgen	$N_pN_qN_r + N_p^2$	$2N_p^2N_qN_r$	$2N_p^2N_qN_r/\sqrt{S}$	$\sqrt{S} \le OI \le \sqrt{S}$	1	–
fdtd-2d	$3N_xN_y + T$	$11N_xN_yT$	$\frac{2}{3\sqrt{3}}N_xN_yT/\sqrt{S}$	$\frac{11}{24}\sqrt{3}\sqrt{S} \le OI \le \frac{33}{2}\sqrt{3}\sqrt{S}$	36	–
floyd-warshall	N^2	$2N^3$	$2N^3/\sqrt{S}$	$\sqrt{S} \le OI \le \sqrt{S}$	1	$8\sqrt{S}$ [1]
gemm	$N_iN_j + N_jN_k + N_iN_k$	$2N_iN_jN_k$	$2N_iN_jN_k/\sqrt{S}$	$\sqrt{S} \le OI \le \sqrt{S}$	1	$\sqrt{S}$ [4]
heat-3d	N^3	$30N^3T$	$\frac{3}{8}\sqrt[3]{2}N^3T/\sqrt[3]{S}$	$\frac{5}{2}\sqrt[3]{S} \le OI \le 40 \cdot 2^{2/3}\sqrt[3]{S}$	$16 \cdot 2^{2/3}$	–
jacobi-1d	N	$6NT$	$\frac{1}{4}NT/S$	$\frac{3}{2}S \le OI \le 24S$	16	$48S$ [3]
jacobi-2d	N^2	$10N^2T$	$\frac{2}{3\sqrt{3}}N^2T/\sqrt{S}$	$\frac{5}{4}\sqrt{S} \le OI \le 15\sqrt{3}\sqrt{S}$	$12\sqrt{3}$	$40\sqrt{2}\sqrt{S}$ [3]
lu	N^2	$\frac{2}{3}N^3$	$\frac{2}{3}N^3/\sqrt{S}$	$\sqrt{S} \le OI \le \sqrt{S}$	1	$8\sqrt{S}$ [1]
ludcmp	N^2	$\frac{2}{3}N^3$	$\frac{2}{3}N^3/\sqrt{S}$	$\sqrt{S} \le OI \le \sqrt{S}$	1	$8\sqrt{S}$ [1]
seidel-2d	N^2	$9N^2T$	$\frac{2}{3\sqrt{3}}N^2T/\sqrt{S}$	$\frac{9}{4}\sqrt{S} \le OI \le \frac{27\sqrt{3}}{2}\sqrt{S}$	$6\sqrt{3}$	–
symm	$\frac{1}{2}M^2 + 2MN$	$2M^2N$	$2M^2N/\sqrt{S}$	$\sqrt{S} \le OI \le \sqrt{S}$	1	$8\sqrt{S}$ [1]
syr2k	$\frac{1}{2}N^2 + 2MN$	$2MN^2$	$MN^2/\sqrt{S}$	$\sqrt{S} \le OI \le 2\sqrt{S}$	2	$8\sqrt{S}$ [1]
syrk	$\frac{1}{2}N^2 + MN$	MN^2	$\frac{1}{2}MN^2/\sqrt{S}$	$\sqrt{S} \le OI \le 2\sqrt{S}$	2	$8\sqrt{S}$ [1]
trmm	$\frac{1}{2}M^2 + MN$	M^2N	$M^2N/\sqrt{S}$	$\sqrt{S} \le OI \le \sqrt{S}$	1	$8\sqrt{S}$ [1]
atax	MN	$4MN$	MN	$4 \le OI \le 4$	1	–
bicg	MN	$4MN$	MN	$4 \le OI \le 4$	1	–
deriche	HW	$32HW$	HW	$\frac{16}{3} \le OI \le 32$	6	–
gemver	N^2	$10N^2$	N^2	$5 \le OI \le 10$	2	–
gesummv	$2N^2$	$4N^2$	$2N^2$	$2 \le OI \le 2$	1	–
mvt	N^2	$4N^2$	N^2	$4 \le OI \le 4$	1	–
trisolv	$\frac{1}{2}N^2$	N^2	$\frac{1}{2}N^2$	$2 \le OI \le 2$	1	–
adi	N^2	$30N^2T$	N^2T	$5 \le OI \le 30$	6	–
durbin	N	$2N^2$	$\frac{1}{2}N^2$	$\frac{2}{3} \le OI \le 4$	6	–
gramschmidt	MN	$2MN^2$	$MN^2/\sqrt{S}$	$1 \le OI \le 2\sqrt{S}$	$2\sqrt{S}$	–
nussinov	$\frac{1}{2}N^2$	$\frac{1}{3}N^3$	$\frac{1}{6}N^3/\sqrt{S}$	$1 \le OI \le 2\sqrt{S}$	$2\sqrt{S}$	–

Tight bound for 14 kernels out of 30

Conclusion

IOLB: First automatic I/O lower bound analysis tool

- Geometric reasoning
- Polyhedral program representation
- State-of-the-art results on a full benchmark suite

Thank you!

References

Grey Ballard et al. "Communication lower bounds and optimal algorithms for numerical linear algebra". In: *Acta Numerica* 23 (2014), pp. 1–155.

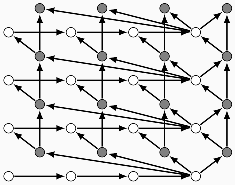
Michael Christ et al. *Communication Lower Bounds and Optimal Algorithms for Programs That Reference Arrays — Part 1*. EECS Technical Report EECS–2013-61. UC Berkeley, May 2013.

Venmugil Elango et al. "On characterizing the data movement complexity of computational DAGs for parallel execution". In: *Proc. of the 26th ACM Symposium on Parallelism in Algorithms and Architectures, SPAA '14, Prague, Czech Republic - June 23 - 25, 2014*. 2014, pp. 296–306.

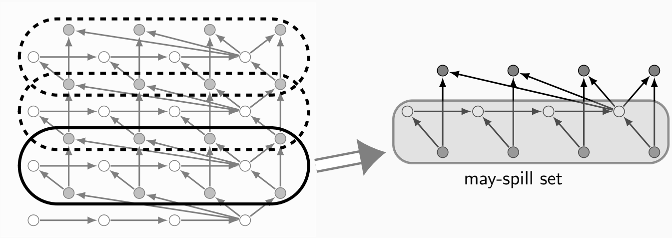
Tyler Michael Smith et al. *A Tight I/O Lower Bound for Matrix Multiplication*. 2019. arXiv: 1702.02017v2.

```
  for(t=0; t<M; t++) {
      s = 0;
      for(i=0; i<N; i++)
S1:         s += A[j];
      for(i=0; i<N; i++)
S2:         A[j] += s;
  }
```

(a) Code



(b) CDAG
for M=4, N=4. White vertices correspond to S1, gray vertices to S2.



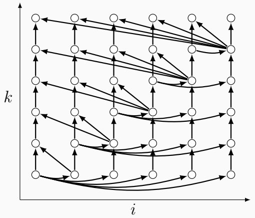
(c) Decomposition of the CDAG

```
Parameters: N;
Input: A[N]; Output: A[N];
for(k=0;k<N;k++)
  for (i=0; i<N; i++)
    A[i] = f(A[i],A[k]);
```

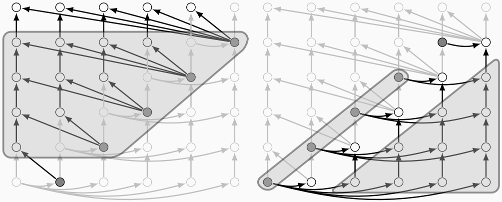
(a) C-like code

```
Parameters: N;
Input: A[N]; Output: S_{N-1}[N];
for (0≤k<N and 0≤i<N)
  if (k==i==0): S_{0,i} = f(A[0],A[0]);
  else if (k==0): S_{0,i} = f(A[i],S_{0,0});
  else if (i<=k): S_{k,i} = f(S_{k-1,i}, S_{k-1,k});
  else if (i>k): S_{k,i} = f(S_{k-1,i}, S_{k,k});
```

(b) Corresponding single assignment form



(c) Corresponding CDAG for N=5. Input nodes A[N] are omited.



(d) Decomposition into two non-interfering sub-CDAGs. Sources are in gray. May-spill sets are encircled.