Estimating the upper bound on arithmetic intensity for a stencil algorithm

Sergey Khilkov

Hipercone Ltd.,
Israel

IMPACT 2025: January 22, 2025

Table of contents

Arithmetic intensity and stencil algorithms

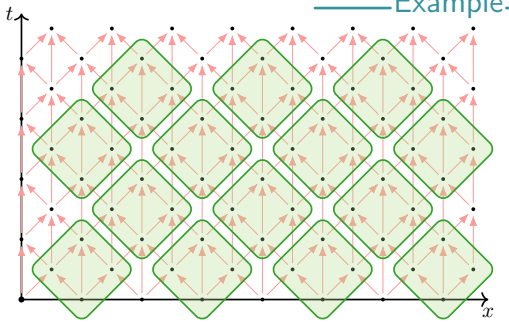
Arithmetic intensity

- $I = O/D$,
  - O is the number of arithmetic operations,
  - D is amount of data traffic to a memory level.
- Has different value for each memory level.
- Is proportional to performance for memory bound problems.
- May be improved with tiling / temporal blocking.
- Is there an upper bound?

Stencil algorithm

- Grid corresponds to a data array, but memory layout is not specified.
- Uniform stencil applied to each point.

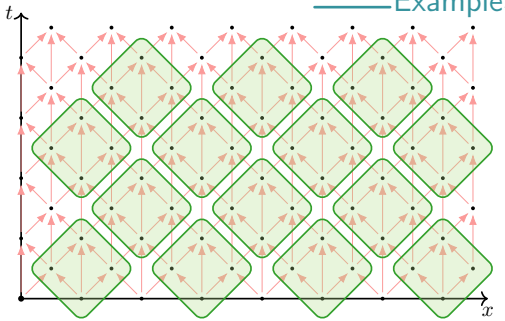
Example



Cache model

- Single cache level.
- Fully-associative.
- Holds all data tile needs for calculation.
- Discards tile data after it has been calculated.

Example



Definition (Minkowski sum)

$A + B = \{a + b | a \in A, b \in B\}$.

Arithmetic intensity for tile T

- $O = |T|$,
  - $|T|$ is the number of points in T.
- $D = |T + S \setminus T|$,
  - S is the stencil.
- $I = \frac{|T|}{|T + S| - |T|}$.

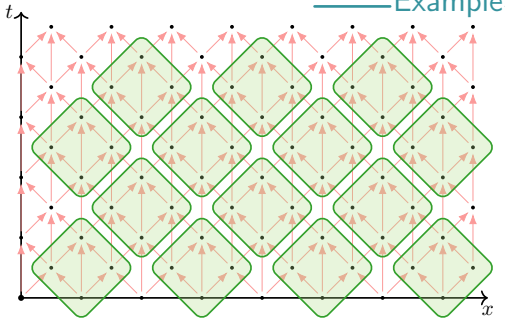
Brunn–Minkowski inequalities in $\mathbb{R}^n$

- Continuous
  $\mathcal{L}(A + B)^{\frac{1}{n}} \geq \mathcal{L}(A)^{\frac{1}{n}} + \mathcal{L}(B)^{\frac{1}{n}}$,
  - $\mathcal{L}(A)$ is Lebesgue measure of A.
- Discrete
  $|A + B|^{\frac{1}{n}} \geq |A|^{\frac{1}{n}} + n!^{-\frac{1}{n}} (|B| - n)^{\frac{1}{n}}$.

Cache model

- Single cache level.
- Fully-associative.
- Holds all $|T+S|-|T|$ data points.
- Discards tile data after it has been calculated.

Example



Arithmetic intensity for tile T

- $I = \dfrac{|T|}{|T+S|-|T|}$.

Brunn–Minkowski inequalities in $\mathbb{R}^n$

- Discrete
  $|A+B|^{\frac{1}{n}} \geq |A|^{\frac{1}{n}} + n!^{-\frac{1}{n}} \left(|B|-n\right)^{\frac{1}{n}}$.

Isoperimetric-like inequality derivation

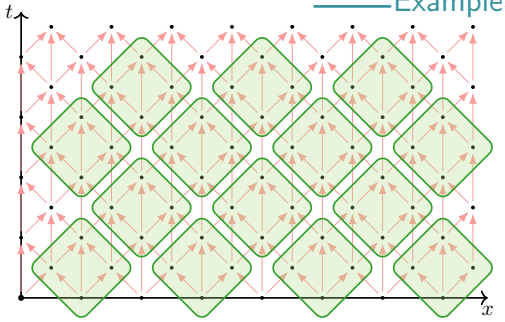
$$|T+S| \geq |T| + \frac{n}{n!^{\frac{1}{n}}} |T|^{\frac{n-1}{n}} \left(|S|-n\right)^{\frac{1}{n}}$$

$$+ \sum_{k=2}^{n} \binom{n}{k} |T|^{n-k} n!^{-\frac{k}{n}} \left(|S|-n\right)^{\frac{k}{n}}.$$

Cache model

- Single cache level.
- Fully-associative.
- Holds all $|T+S|-|T|$ data points.
- Discards tile data after it has been calculated.

Example



Arithmetic intensity for tile T

- $I = \dfrac{|T|}{|T+S|-|T|}$.

Brunn–Minkowski inequalities in $\mathbb{R}^n$

- Discrete
  $|A+B|^{\frac{1}{n}} \geq |A|^{\frac{1}{n}} + n!^{-\frac{1}{n}} \left(|B|-n\right)^{\frac{1}{n}}$.

Isoperimetric-like inequality

$$\frac{n!}{n^n}\frac{1}{|S|-n} \geq \frac{I^{n-1}}{|T+S|-|T|}.$$

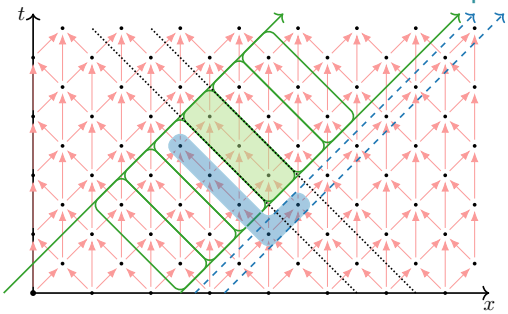
Example

- Isoperimetric $\frac{1}{4} \geq \frac{|T|}{(|T+S|-|T|)^2}$.
- For Diamond Tile $\frac{k^2}{(2k+1)^2} \xrightarrow[k\to\infty]{} \frac{1}{4}$.

Cache model

- Single cache level.
- Fully-associative.
- ~~Holds all data tile needs for calculation.~~
- ~~Discards tile data once it has been calculated.~~

Example

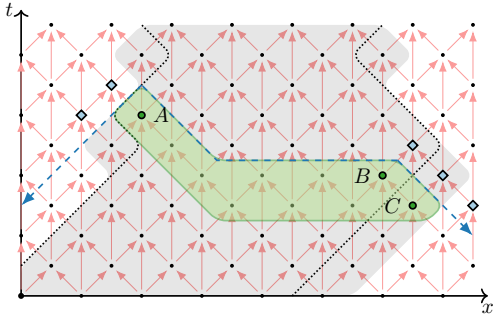


- Uncached loads occur only through solid arrows.
- Cache size D_{cache} is larger than cross-section.
- For this example: $I = D_{\text{cache}}$.

Cache model

- Single cache level.
- Fully-associative.

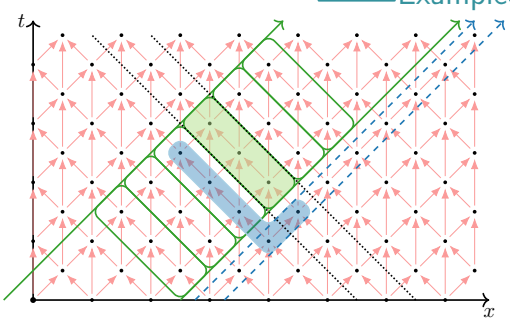
Continuous cache fragment



Conjecture

For a stencil algorithm in n-dim plus time: $I \leq C \cdot \sqrt[n]{D_{\text{cache}}}$.

Example



- For this example: $I = D_{\text{cache}}$.

Isoperimetric-like inequality

$$\frac{(n+1)!}{(n+1)^{n+1}} \frac{|T|}{|S| - n - 1} \geq I^{n+1}.$$

How to estimate C?

1. Describe a model of continuous tilings.

2. Show that every discrete tiling corresponds to at least one continuous tiling.

3. Use the continuous model to obtain an intensity limit using an isoperimetric-like inequality.

Geometric locality model

Atomic tiling

- Tiling is a general tessellation / honeycomb.
- Tiling provides a way to split big task in smaller ones.
- Atomic equals valid.

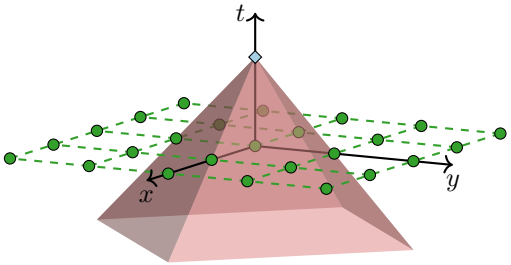
Underlying space

- Vector space with a norm.
- Dependencies described with the cone Cone_d (similar to light cone in Minkowski space).
- Base of the dependence cone is a metric ball corresponding to a norm.

Restrictions on the stencil

- Full-dimensional
- Bounded
- Centrally symmetrical (space part)
- Convex

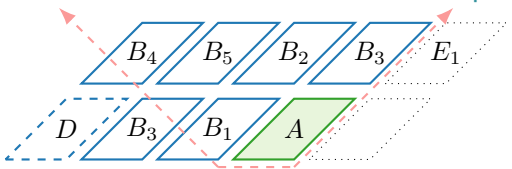
Example



Definition

If the interior of tile B intersects with the dependence cone of tile A, then A depends on B.

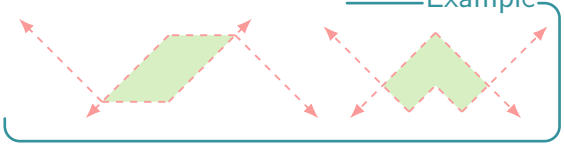
Example



Definition (Conoid)

We will call a set A conoid iff $A = \overline{A + \mathrm{Int}\,(-\mathrm{Cone}_d) \cap A + \mathrm{Int}\,(\mathrm{Cone}_d)}$.

Example



Theorem

Every atomic tiling consists of conoids.

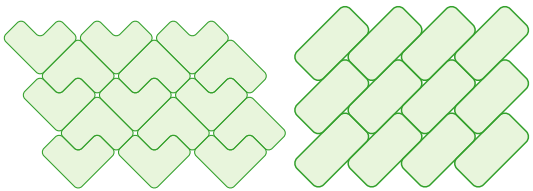
Proof

For any given shape other than a conoid, there is other tile, which simultaneously depends and is dependent on it. □

Note

Tiling which consists of conoids is not necessarily atomic.

Example



Proposition

Tiling with exactly two conoids is atomic. We call such tilings ***binary****.*

Proposition

Intersection of two atomic tilings is atomic.

Theorem

Tiling is atomic iff it can be produced by an intersection of binary tilings.

Proof

- If case
  - Direct consequence of propositions.
- Only if case
  - For atomic tiling there exist a total order on tiles, which is compatible with dependencies.
  - For any tile T we can split the tiling into two sets:
    - $T_< =$ every tile before T,
    - $T_\geq = T$ and every tile after it.
  - $\{T_<, T_\geq\}$ is the atomic tiling.
  - Intersection of all such tilings gives the initial tiling. □

Theorem

There exists an atomic tiling corresponding to any given valid discrete tiling.

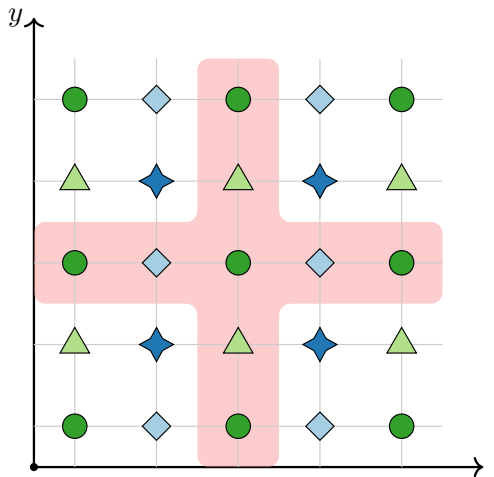
Proof sketch

1. There exists an atomic tiling:
   - 1.1 each tile contains at most one point of the lattice,
   - 1.2 dependence relation on tiles which contain points equals to dependence relation on points.
2. It is possible to join tiles in atomic tiling to get another atomic tiling.
3. The set of rules governing such joins is the same as for joining tiles in a valid discrete tiling to get a valid discrete tiling.
4. A valid discrete tiling can be constructed from points using valid joining rules.
5. The same joins may be performed for the atomic tiling in the step 1. □

Example (Obtaining the limit)

For the example scheme: $I \leq D_{\text{cache}}$.

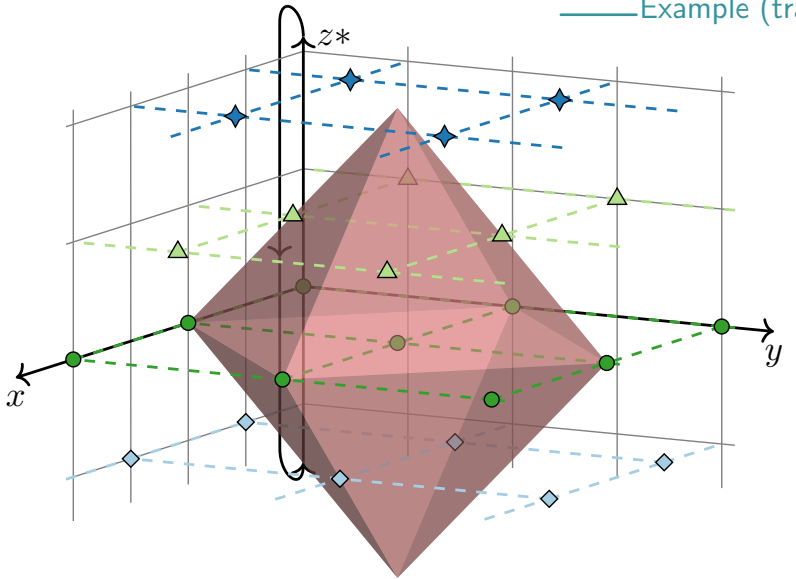
Example



Transformation

- z is a periodic axis, $(z = -2) \equiv (z = 2)$.
- for subgrid $(0, 0)$, we add $z = 0$,
- for subgrid $(1, 0)$, we add $z = -1$,
- for subgrid $(0, 1)$, we add $z = 1$,
- for subgrid $(1, 1)$, we add $z = 2$.

Example (transformed)



Initial kernel

```
for(int t=0; t<T-2; t++) {
  for(int i=2; i<N-3; i++) {
    for(int j=2; j<M-3; j++) {
      A[t+1][i][j] = f(A[t][i-2][j], A[t][i-1][j], A[t][i][j], A[t][i+1][j],
        A[t][i+2][j], A[t][i][j-2], A[t][i][j-1], A[t][i][j+1], A[t][i][j+2]);
} } }
```

Transformed kernel

```
for(int t=0; t<T-2; t++) {
  for(int i=2; i<N-3; i++) {
    for(int j=2; j<M-3; j++) {
      for(int p=0; p<2; p++){
        for(int q=0; q<2; q++){
          if ((i%2==p) && (j%2==q)) {
            A[t+1][i][j] = f(A[t][i-2][j], A[t][i-1][j], A[t][i][j],
               A[t][i+1][j], A[t][i+2][j], A[t][i][j-2], A[t][i][j-1],
             A[t][i][j+1], A[t][i][j+2]);
} } } } } }
```

Initial kernel

```
for(int t=0; t<T-2; t++) {
  for(int i=2; i<N-3; i++) {
    for(int j=2; j<M-3; j++) {
      S1;
} } }
```

Transformed kernel

```
for(int t=0; t<T-2; t++) {
  for(int i=2; i<N-3; i++) {
    for(int j=2; j<M-3; j++) {
      for(int p=0; p<2; p++){
        for(int q=0; q<2; q++){
          if ((i%2==p) && (j%2==q)) {
            S1;
} } } } } }
```

Old dependence cone

$\pm i \pm j + 2t \geq 0.$

New dependence cone

- $\pm i \pm j \pm p \pm q + 2t \geq 0.$
- All tiling hyperplanes allowed by initial loop are also allowed by the new one.

Example (new allowed tiling)

- Tiling hyperplanes:
  - $i + p + 2t = 0,$
  - $j + q + 2t = 0,$
  - $-i - j - p - q + 2t = 0,$

Conclusion

- Geometric inequalities help to estimate the upper bound on arithmetic intensity of a stencil algorithm.
- We propose a conjecture that a bound has a form $I \leq C \sqrt[n]{D_{\text{cache}}}$.
- The continuous geometric locality model was introduced to simplify estimations of C in the asymptotic limit.
- The opportunity to lift the model restrictions was demonstrated on the example of non-linear transformation.
- Same transformation may be useful to extend the space of achievable tilings in polyhedral model.

Thank you!