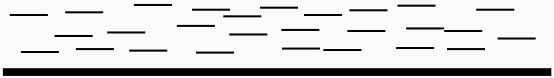
Asymptotically optimal minimizers schemes

Guillaume Marçais, Dan DeBlasio, Carl Kingsford

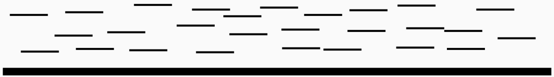
Carnegie Mellon University

Computing read overlaps

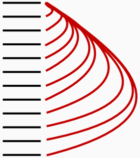


Roberts, *et al.* (2004). Reducing storage requirements for biological sequence comparison.

Computing read overlaps



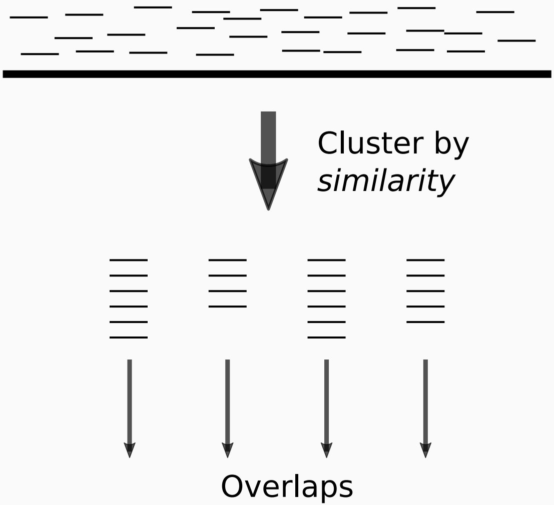
Roberts, *et al.* (2004). Reducing storage requirements for biological sequence comparison.



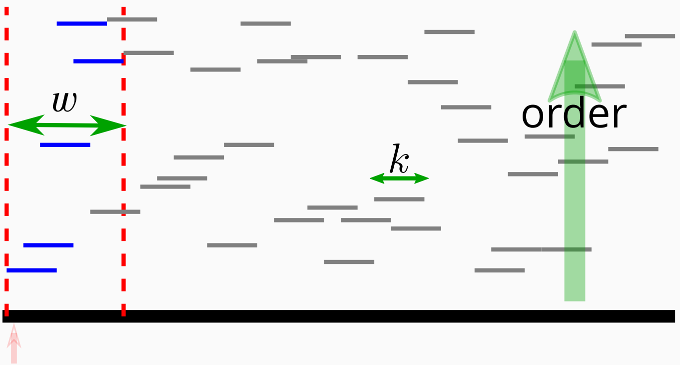
$O(n^2)$ alignments

Computing read overlaps

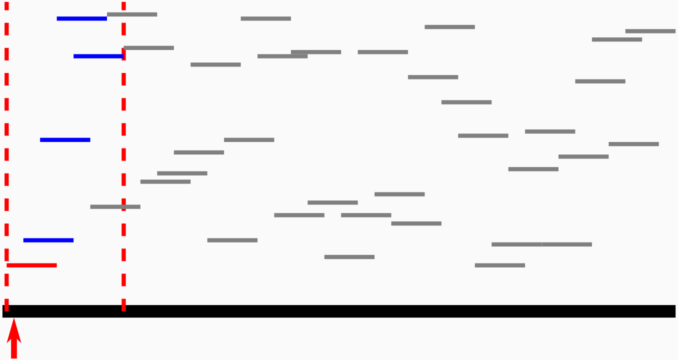
Roberts, *et al.* (2004). Reducing storage requirements for biological sequence comparison.



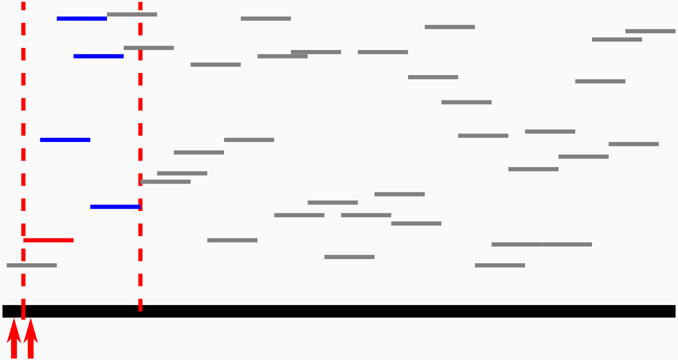
Computing minimizers



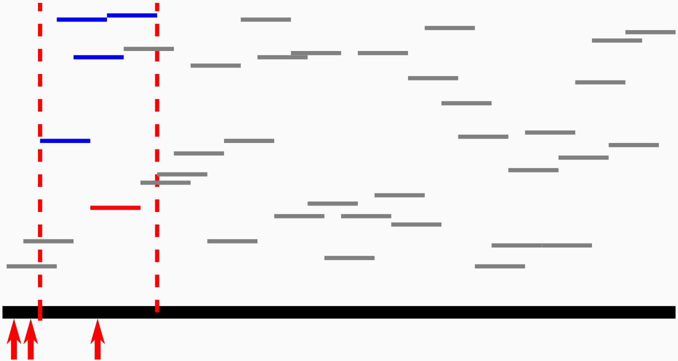
Computing minimizers



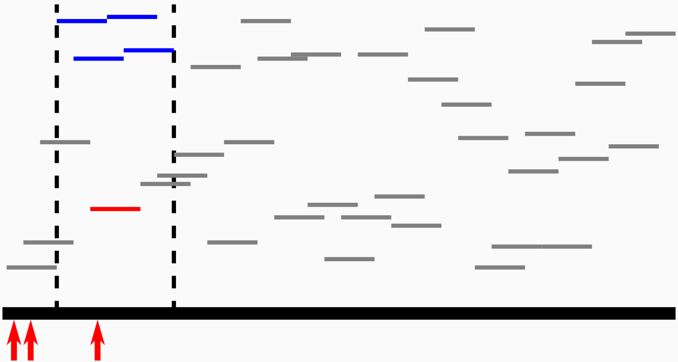
Computing minimizers



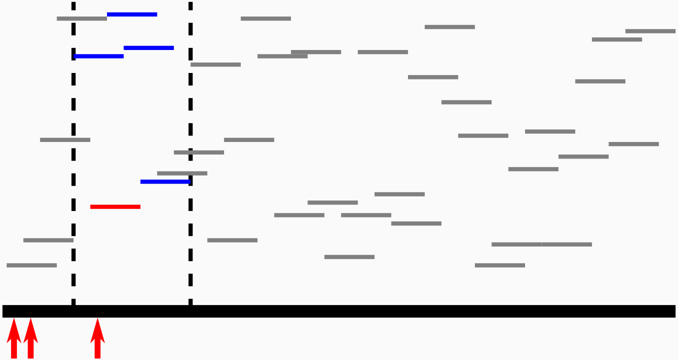
Computing minimizers



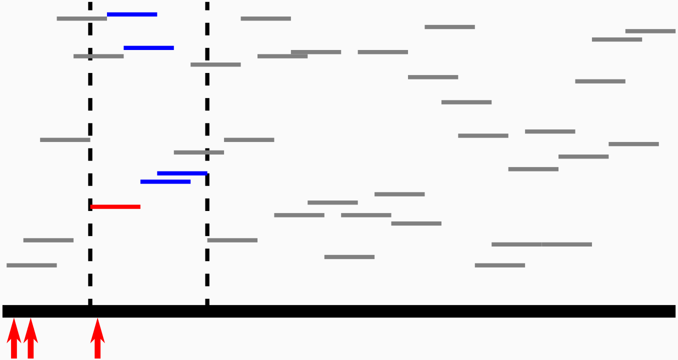
Computing minimizers



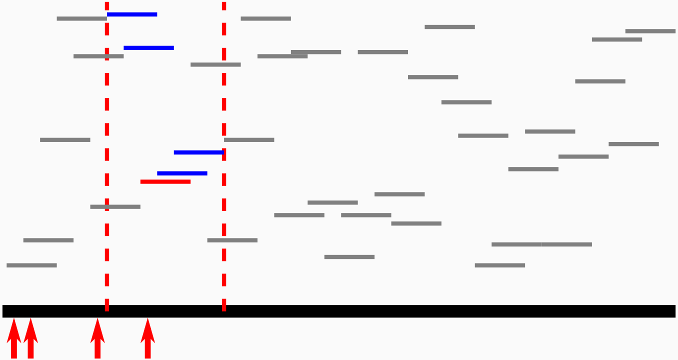
Computing minimizers



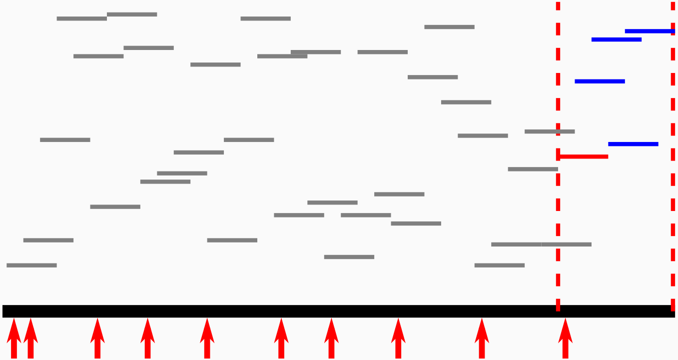
Computing minimizers



Computing minimizers



Computing minimizers



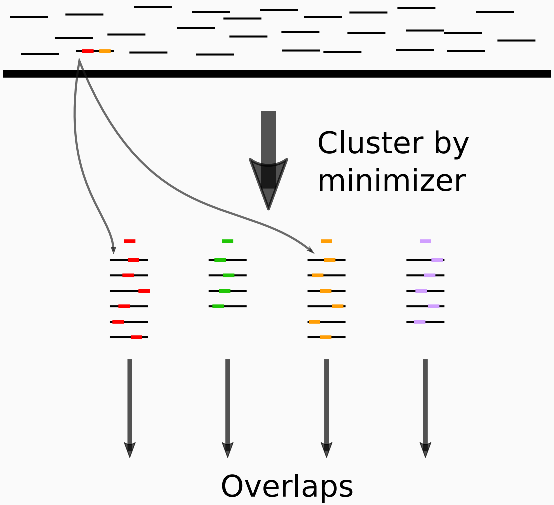
Minimizers definition and properties

Minimizers (k, w, o)
In each window of w consecutive k-mers, select the smallest k-mer according to order o.

1. **No large gap**: distance between selected k-mers is $\leq w$
2. **Deterministic**: two strings matching on w consecutive k-mers select the same minimizer

Computing read overlaps

1. **No large gap**: no sequence ignored
2. **Deterministic**: reads with overlap in same bin

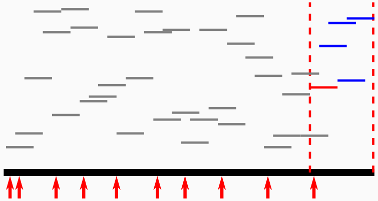


Many applications of minimizers

- **UMDOverlapper (Roberts, 2004)**: bin sequencing reads by shared minimizers to compute overlaps
- **MSPKmerCounter (Li, 2015), KMC2 (Deorowicz, 2015), Gerbil (Erber, 2017)**: bin input sequences based on minimizer to count *k*-mers in parallel
- **SparseAssembler (Ye, 2012), MSP (Li, 2013), DBGFM (Chikhi, 2014)**: reduce memory footprint of de Bruijn assembly graph with minimizers
- **SamSAMi (Grabowski, 2015)**: sparse suffix array with minimizers
- **MiniMap (Li, 2016), MashMap (Jain, 2017)**: sparse data structure for sequence alignment
- **Kraken (Wood, 2014)**: taxonomic sequence classifier

- **Schleimer *et al.* (2003)**: winnowing

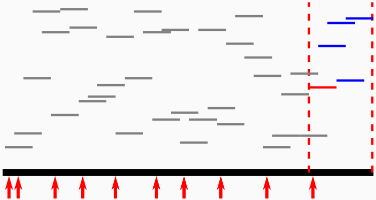
Improving minimizers by lowering density

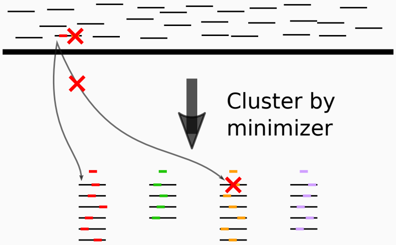


Density
Density of a scheme is the expected proportion of selected k-mer in a random sequence:

$$d = \frac{\#\text{ of selected } k\text{-mers}}{\text{length of sequence}}$$

Improving minimizers by lowering density





Density
Density of a scheme is the expected proportion of selected *k*-mer in a random sequence:

$$d = \frac{\#\text{ of selected } k\text{-mers}}{\text{length of sequence}}$$

Lower density
$\Longrightarrow$ smaller bins
$\Longrightarrow$ less computation

Minimizers density minimizing problem

For fixed k and w:

- Properties “No large gap” & “Deterministic” unaffected by order
- Density changes with ordering o
- Lower density $\implies$ sparser data structures and/or less computation
- Benefit existing and new applications

Density minimization problem
For fixed w, k, find k-mer **order** o giving the lowest expected density

Minimizers density minimizing problem

For fixed k and w:

- Properties “No large gap” & “Deterministic” unaffected by order
- Density changes with ordering o
- Lower density $\implies$ sparser data structures and/or less computation
- Benefit existing and new applications

Density minimization problem
For fixed w, k, find k-mer **order** o giving the lowest expected density

Density and density factor trivial bounds

$$\underbrace{\frac{1}{w}}_{\text{Pick every other } w \text{ } k\text{-mer}} \leq d \leq \overbrace{1}^{\text{Pick every } k\text{-mer}}$$

Random order *usual* expected density $d = \frac{2}{w+1}$

$$1 + \frac{1}{w} \leq \text{df} = (w+1) \cdot d \leq w+1$$

Random order usual expected *density factor* $\text{df} = 2$

Density and density factor trivial bounds

$$\underbrace{\frac{1}{w}}_{\text{Pick every other } w \text{ } k\text{-mer}} \leq d \leq \overbrace{1}^{\text{Pick every } k\text{-mer}}$$

Random order *usual* expected density $d = \frac{2}{w+1}$

$$1 + \frac{1}{w} \leq \mathrm{df} = (w+1) \cdot d \leq w + 1$$

Random order usual expected *density factor* $\mathrm{df} = 2$

Schleimer's bound does not apply in general

$$d \geq \frac{1.5 + \frac{1}{2w}}{w + 1} \qquad \text{(Schleimer \textit{et al.})}$$

Schleimer's bound does not apply in general

$$d \geq \frac{1.5 + \frac{1}{2w}}{w + 1}$$ (Schleimer *et al.*)

Applies only if $w \gg k$, or for *random* orders

Schleimer's bound does not apply in general

$$d \geq \frac{1.5 + \frac{1}{2w}}{w+1}$$ (Schleimer *et al.*)

Applies only if $w \gg k$, or for *random* orders

$$d \geq \frac{1.5 + \frac{1}{2w} + \max\left(0, \lfloor \frac{k-w}{w} \rfloor\right)}{w+k}$$

Valid for **any** k, w and **any** order

Schleimer's bound does not apply in general

$$d \geq \frac{1.5 + \frac{1}{2w}}{w+1} \qquad \text{(Schleimer } \textit{et al.}\text{)}$$

Applies only if $w \gg k$, or for *random* orders

$$d \geq \frac{1.5 + \frac{1}{2w} + \max\left(0, \lfloor\frac{k-w}{w}\rfloor\right)}{w+k} \qquad \left(\xrightarrow[k\to\infty]{} \frac{1}{w}\right)$$

Valid for **any** k, w and **any** order

Asymptotic behavior in k and w

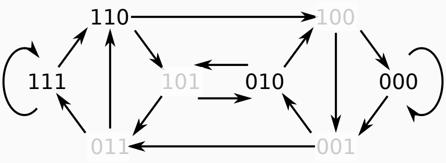
What is the best ordering possible when:

- w is fixed and $k \to \infty$
- k is fixed and $w \to \infty$

A universal set defines an ordering

Universal set
A set M of k-mers that intersects every path of w nodes in the de Bruijn graph of order k.

- $w = 2 \implies M$ is a vertex cover
- From M, get order with density $d \leq \frac{|M|}{\sigma^k}$



A universal set defines an ordering

Universal set
A set M of k-mers that intersects every path of w nodes in the de Bruijn graph of order k.

- $w = 2 \implies M$ is a vertex cover
- From M, get order with density $d \leq \frac{|M|}{\sigma^k}$

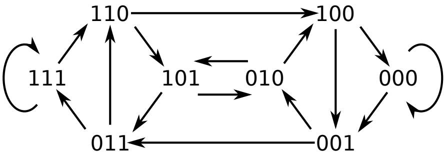
Universal set of size $\frac{\sigma^k}{w}$

$$\Downarrow$$

Order with density $\frac{1}{w}$

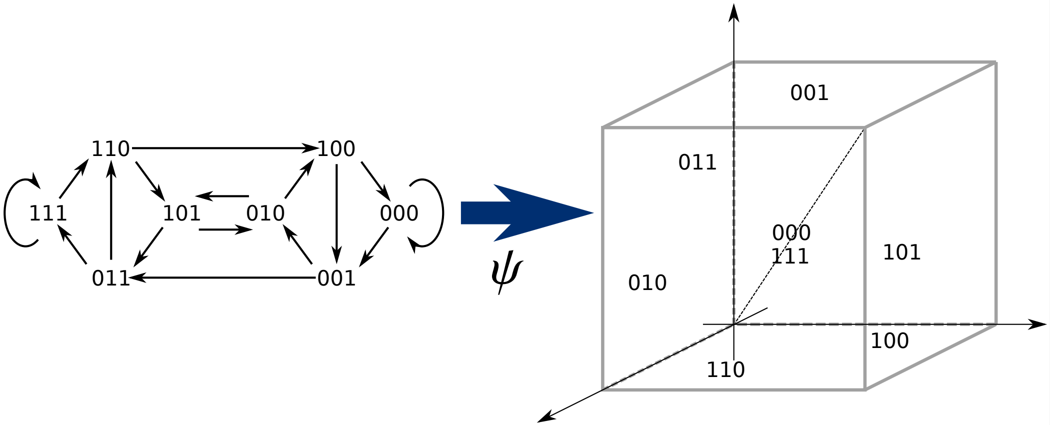
Creating a universal set, $k = 3, w = 3$, algorithm overview

Start with a de Bruijn graph



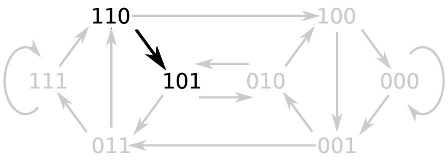
Creating a universal set, $k = 3, w = 3$, algorithm overview

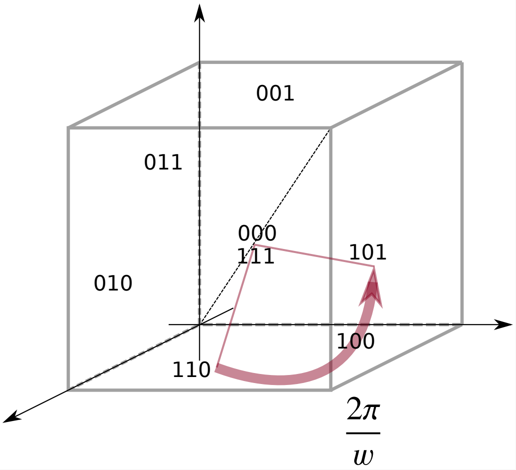
Embed into a w dimensional space using ψ



Creating a universal set, $k = 3, w = 3$, algorithm overview

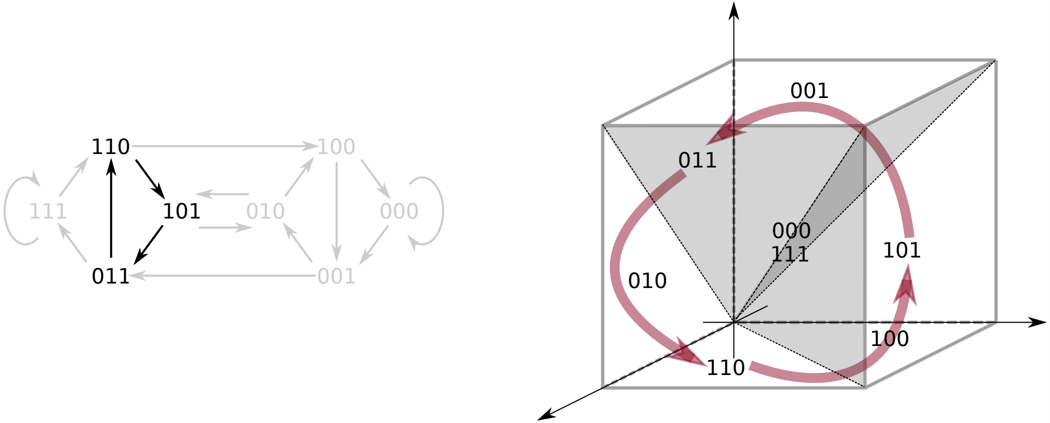
An edge correspond (almost) to a rotation by $2\pi/w$





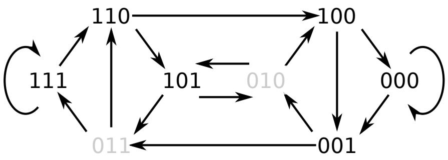
Creating a universal set, $k = 3, w = 3$, algorithm overview

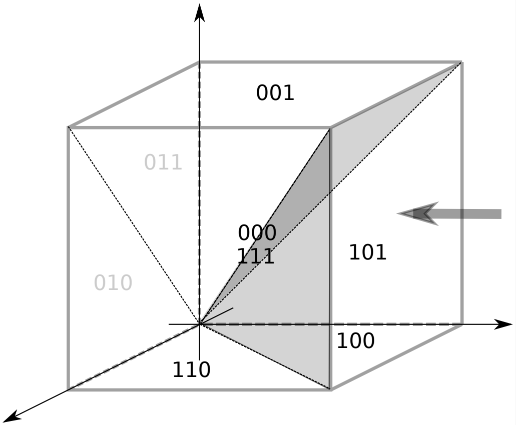
After w edges return to same sub-volume



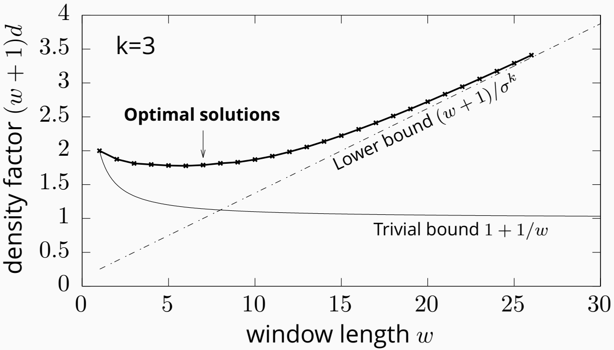
Creating a universal set, $k = 3, w = 3$, algorithm overview

Pick *k*-mers in the highlighted "wedge"





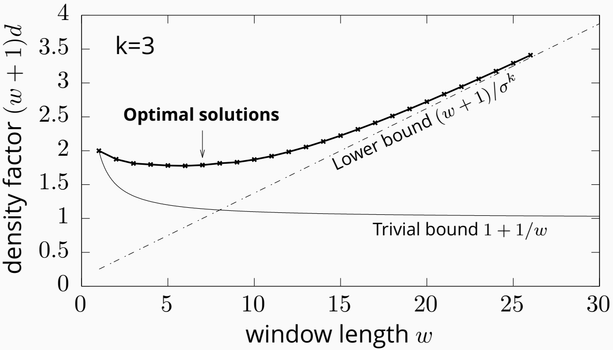
Asymptotic behavior in w



$$d \geq \frac{1}{\sigma^k}, \quad \mathrm{df} \geq \frac{w+1}{\sigma^k}$$

Density factor is $\theta(w)$, not constant

Asymptotic behavior in w



$$d \geq \frac{1}{\sigma^k}, \quad \mathrm{df} \geq \frac{w+1}{\sigma^k}$$

Density factor is $\theta(w)$, not constant

Summary

Asymptotic behavior of minimizers is fully characterized:

- Minimizers scheme is optimal for large k: $d \xrightarrow[k\to\infty]{} \frac{1}{w}$
- Minimizers scheme is not optimal for large w: $\mathrm{df} = \theta(w)$
- Tighter lower bound

$$d \geq \frac{1.5 + \frac{1}{2w} + \max\left(0, \lfloor\frac{k-w}{w}\rfloor\right)}{w+k}$$

- Comparison between k-mers take $O(k)$

Future work

- Local scheme: $f : \Sigma^{w+k-1} \to [1, w]$
- Local schemes *might* be optimal for large w

Carl Kingsford group:

Dan DeBlasio
Heewook Lee
Mingfu Shao
Brad Solomon
Natalie Sauerwald
Cong Ma
Hongyu Zheng
Laura Tung
Postdoc position open

Carnegie Mellon University

GORDON AND BETTY MOORE FOUNDATION
GBMF4554

NSF
CCF-1256087
CCF-1319998

NATIONAL INSTITUTES OF HEALTH
R01HG007104
R01GM122935

The Shurl and Kay Curci Foundation