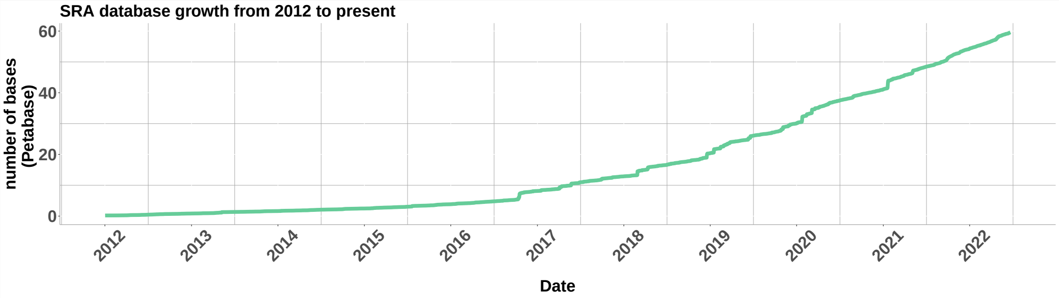
Fractional Hitting Sets & SuperSampler

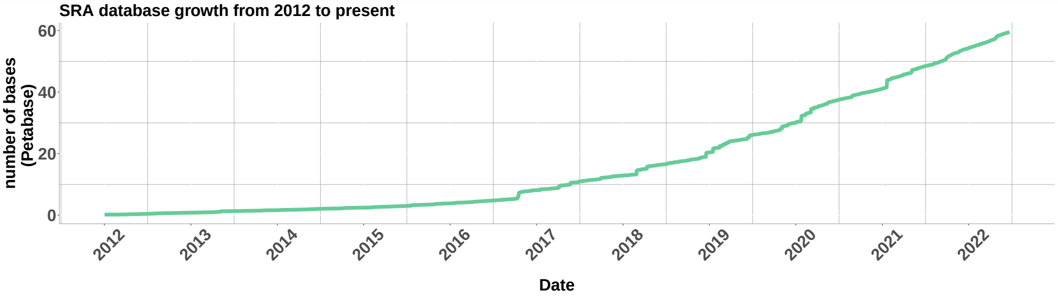
Efficient and Lightweight Genomic Data Sketching

Timothé Rouzé, Igor Martayan, Camille Marchet & Antoine Limasset

July 25, 2023

CRIStAL Centre de Recherche en Informatique, Signal et Automatique de Lille
cnrs
Université de Lille
ENS rennes


SRA database growth from 2012 to present
number of bases (Petabase)
0
20
40
60
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
Date

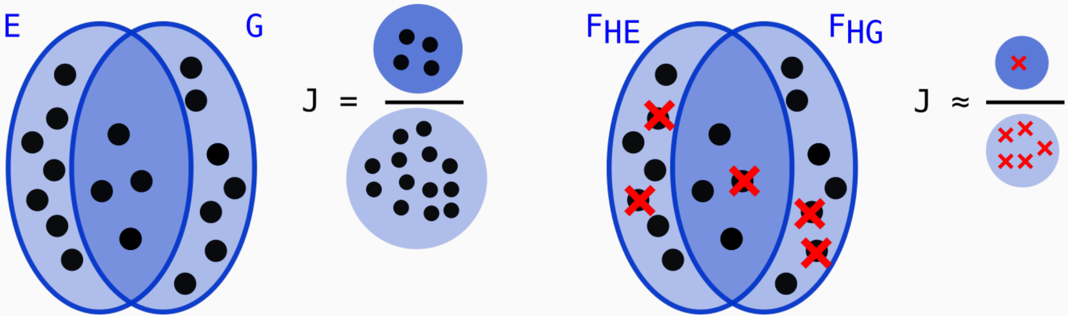


Problems

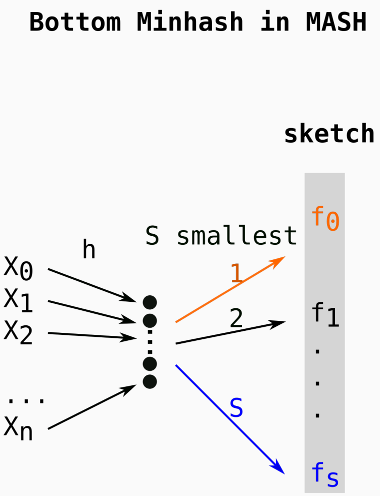
- keeps increasing
- large scale analysis not possible

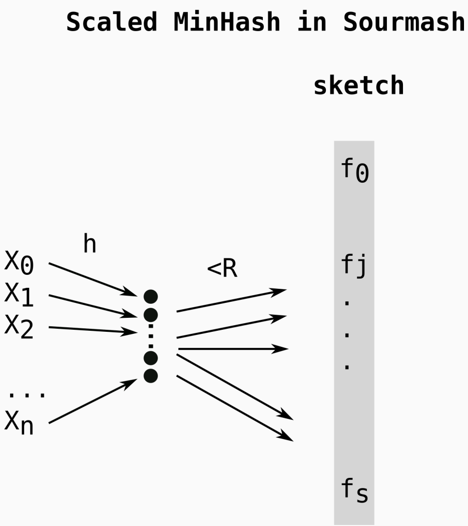
Preliminaries

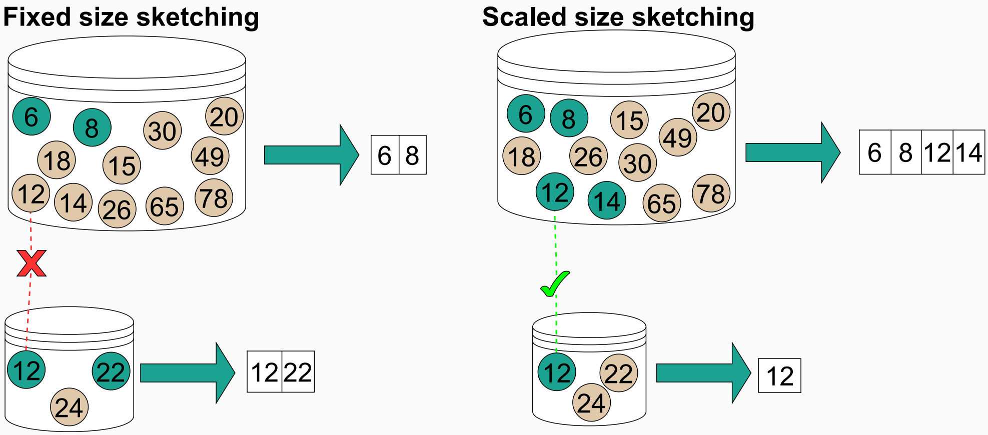
Jaccard index



Sketching


Bottom Minhash in MASH
sketch
S smallest
h
X0
X1
X2
...
Xn
1
2
S
f0
f1
.
.
.
fs


Scaled MinHash in Sourmash
sketch
h
X0
X1
X2
...
Xn
<R
f0
fj
.
.
.
fs


Fixed size sketching
6
8
30
20
18
15
49
12
14
26
65
78
6 8
12
22
24
12 22
Scaled size sketching
6
8
15
20
49
18
26
30
12
14
65
78
6 8 12 14
12
22
24
12

Theoretical background

Minimizers & super-k-mers

Minimizer

smallest m-mer of a k-mer according to some order (e.g. lexicographical)

width parameter: $w = k - m + 1$

sequence CTGAAATGCACATTT

k-mers (k=6, m=3)

CTG**AAA**
T G**AAA**T
G**AAA**TG
AAATGC
AATGCA
ATGCAC
TGC**ACA**
GC**ACA**T
C**ACA**TT
ACATTT

super-k-mers

Minimizers & super-k-mers

Minimizer

smallest m-mer of a k-mer according to some order (e.g. lexicographical)

width parameter: $w = k - m + 1$

Super-k-mer

run of consecutive k-mers sharing the same minimizer

sequence CTGAAATGCACATTT

k-mers (k=6, m=3)

CTG**AAA**
T**GAAA**T
GAAATG
AAATGC
AATGCA
ATGCAC
TGC**ACA**
GC**ACA**T
C**ACA**TT
ACATTT

super-k-mers

We use minimizers as a footprint for selecting super-k-mers

We want a sparse minimizer set

Density

$$d = \frac{\#\text{selected minimizers}}{\#m\text{-mers}}$$

Density of minimizers

We want a sparse minimizer set

Density

$$d = \frac{\#\text{selected minimizers}}{\#m\text{-mers}}$$

sequence	TGTTGTGCTATT
k-mers (k=6, m=3)	T**GTT**GT
	GTT**GTG**
	TT**GTG**C
	TGT**GCT**
	GTG**CTA**
	TG**CTA**T
	GCT**ATT**
selected minimizers	.*..*.**.*

high density

(lexicographical order)

We want a sparse minimizer set

Density

$$d = \frac{\#\text{selected minimizers}}{\#m\text{-mers}}$$

	high density (lexicographical order)	low density (TGT < CTA < ...)
sequence	TGTTGTGCTATT	TGTTGTGCTATT
k-mers (k=6, m=3)	T**GTT**GT	TGT**TGT**
	GTT**GTG**	GT**TGT**G
	TT**GTG**C	T**TGT**GC
	TGT**GCT**	**TGT**GCT
	GTG**CTA**	GTG**CTA**
	TG**CTA**T	TG**CTA**T
	GCT**ATT**	G**CTA**TT
selected minimizers	.*..*.**.*	...*....*..

We want a sparse minimizer set

Density

$$d = \frac{\#\text{selected minimizers}}{\#m\text{-mers}}$$

sequence	TGTTGTGCTATT
k-mers (k=6, m=3)	T**GTT**GT
	GTT**GTG**
	TT**GTG**C
	TGT**GCT**
	GTG**CTA**
	TG**CTA**T
	GCT**ATT**
selected minimizers	.*..*.**.*

high density

(lexicographical order)

sequence	TGTTGTGCTATT
k-mers (k=6, m=3)	TGT**TGT**
	GT**TGT**G
	T**TGT**GC
	TGTGCT
	GTG**CTA**
	TG**CTA**T
	G**CTA**TT
selected minimizers	...*....*..

low density

(TGT < CTA < ...)

low density $\iff$ long super-k-mers

We want a sparse minimizer set

Density

$$d = \frac{\#\text{selected minimizers}}{\#m\text{-mers}}$$

Optimal density: $d = 1/w$

When using a random order, the expected density is $\frac{2}{w+1}$

	high density (lexicographical order)	low density (TGT < CTA < ...)
sequence	TGTTGTGCTATT	TGTTGTGCTATT
k-mers (k=6, m=3)	T**GTT**GT	TGT**TGT**
	GTT**GTG**	GT**TGT**G
	TT**GTG**C	T**TGT**GC
	TGT**GCT**	**TGT**GCT
	GTG**CTA**	GTG**CTA**
	TG**CTA**T	TG**CTA**T
	GCT**ATT**	G**CTA**TT
selected minimizers	.*..*.**.*	...*...*..

low density $\iff$ long super-k-mers

Universal Hitting Sets & Density lower bound

Universal Hitting Set (UHS)

set S of m-mers s.t. every run of w consecutive m-mers has $\geqslant 1$ element in S

$$1 \to 2 \to \cdots \to w-1 \to w$$

Universal Hitting Sets & Density lower bound

Universal Hitting Set (UHS)

set S of m-mers s.t. every run of w consecutive m-mers has $\geqslant 1$ element in S

Density lower bound

In any UHS, the density is $\geqslant \frac{1.5}{w+1}$ (i.e. the density factor is $\geqslant 1.5$)

Universal Hitting Set (UHS)

set S of m-mers s.t. every run of w consecutive m-mers has $\geqslant 1$ element in S

$$1 \longrightarrow 2 \longrightarrow \cdots \longrightarrow w-1 \longrightarrow w$$

Density lower bound

In any UHS, the density is $\geqslant \frac{1.5}{w+1}$ (i.e. the density factor is $\geqslant 1.5$)

Can we cross this lower bound by relaxing some constraints?

Fractional Hitting Sets

Instead of covering every k-mer, we cover a fraction f of them

Fractional Hitting Set (FHS)

set S of m-mers s.t. every run of w consecutive m-mers has $\geqslant 1$ element in S with probability $\geqslant f$

Instead of covering every k-mer, we cover a fraction f of them

Fractional Hitting Set (FHS)

set S of m-mers s.t. every run of w consecutive m-mers has $\geqslant 1$ element in S with probability $\geqslant f$

In practice, we select minimizers smaller than a certain threshold t

$$t = \left[1 - (1 - f)^{1/w}\right] \cdot 4^m$$

minimizers $\leqslant t$ are called small minimizers

Density upper bound

Given a covering fraction f, assuming $m > (3+\varepsilon)\log_4 w$,

$$d \leqslant \frac{2f}{w+1} + o(1/w)$$

Density upper bound

Given a covering fraction f, assuming $m > (3+\varepsilon)\log_4 w$,

$$d \leqslant \frac{2f}{w+1} + o(1/w)$$

- $\oplus$ simple, consistent with known results for $f = 1$

Density upper bound

Given a covering fraction f, assuming $m > (3 + \varepsilon) \log_4 w$,

$$d \leqslant \frac{2f}{w+1} + o(1/w)$$

- $\oplus$ simple, consistent with known results for $f = 1$
- $\ominus$ not very meaningful as $f \to 0$
  (since most k-mers are not covered)

Density upper bound

Given a covering fraction f, assuming $m > (3+\varepsilon)\log_4 w$,

$$d \leqslant \frac{2f}{w+1} + o(1/w)$$

- ⊕ simple, consistent with known results for $f = 1$
- ⊖ not very meaningful as $f \to 0$
  (since most k-mers are not covered)

Is there a more meaningful metric?

Restricted density upper bound

Given a covering fraction f, assuming $m > (3+\varepsilon)\log_4 w$,
when restricting to k-mers containing small minimizers,

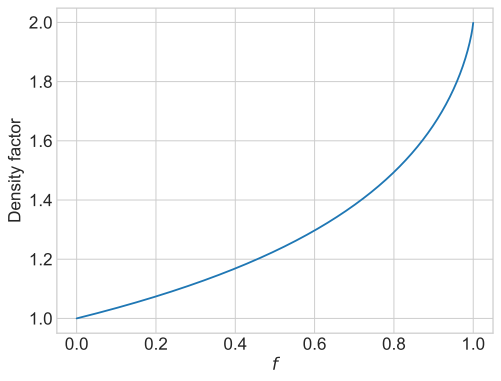
$$d \leqslant 2 \cdot \frac{f + (1-f)\ln(1-f)}{f^2(w+1)} + o(1/w)$$

Restricted density upper bound

Given a covering fraction f, assuming $m > (3+\varepsilon)\log_4 w$, when restricting to k-mers containing small minimizers,

$$d \leqslant 2 \cdot \frac{f + (1-f)\ln(1-f)}{f^2(w+1)} + o(1/w)$$

- below the $\frac{1.5}{w+1}$ barrier for $f \leqslant 0.8$
- approaches optimal density as $f \to 0$



Proportion of maximal super-k-mers

The average proportion of maximal super-k-mers is

$$\left[\left(1-\frac{1}{w}\right)\frac{f}{1+f}\right]^2+\frac{1-f(1-2/w)}{1+f}$$

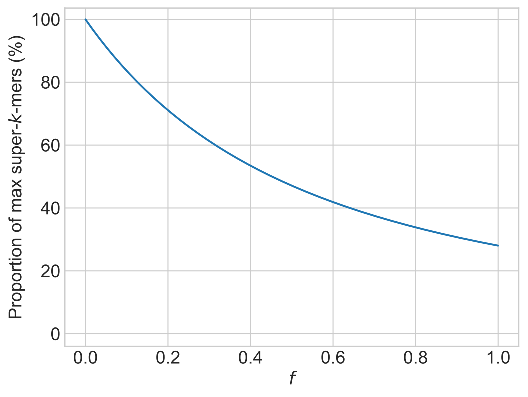
Proportion of maximal super-k-mers

Proportion of maximal super-k-mers

The average proportion of maximal super-k-mers is

$$\left[\left(1-\frac{1}{w}\right)\frac{f}{1+f}\right]^2+\frac{1-f(1-2/w)}{1+f}$$

(for $w = 17$)

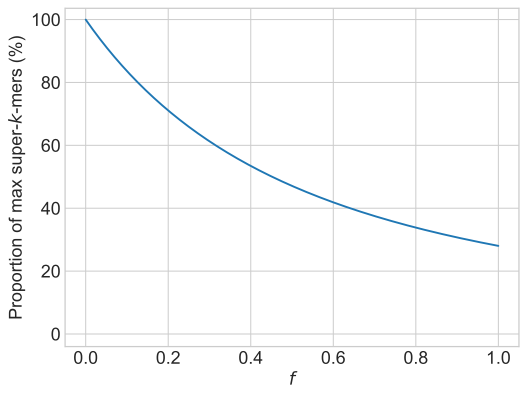


Proportion of maximal super-k-mers

The average proportion of maximal super-k-mers is

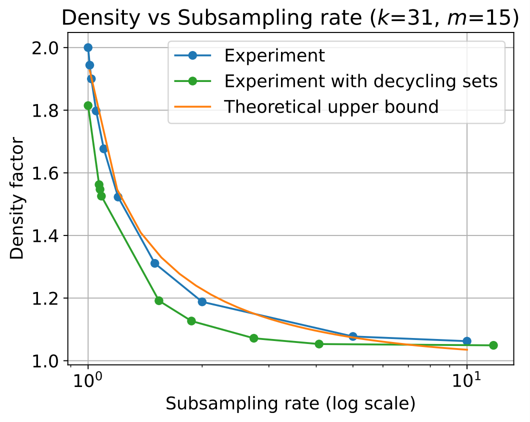
$$\left[\left(1-\frac{1}{w}\right)\frac{f}{1+f}\right]^2+\frac{1-f(1-2/w)}{1+f}$$

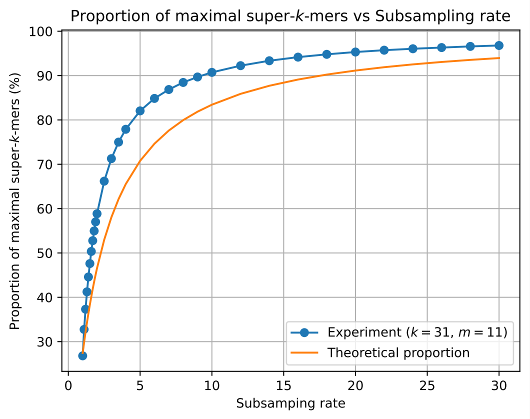
(for $w = 17$)



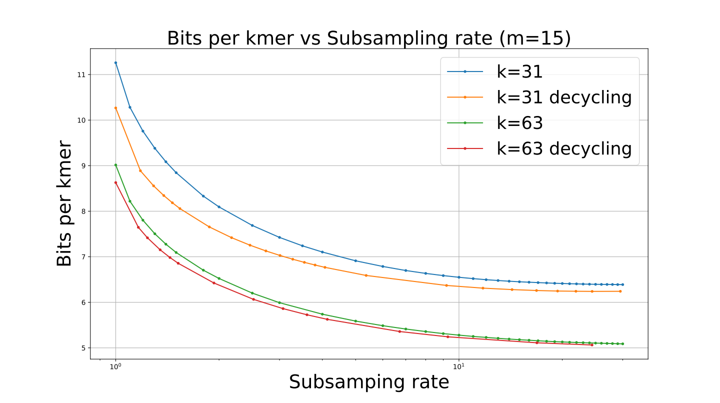
How accurate is it in practice?

Comparison with experimental results



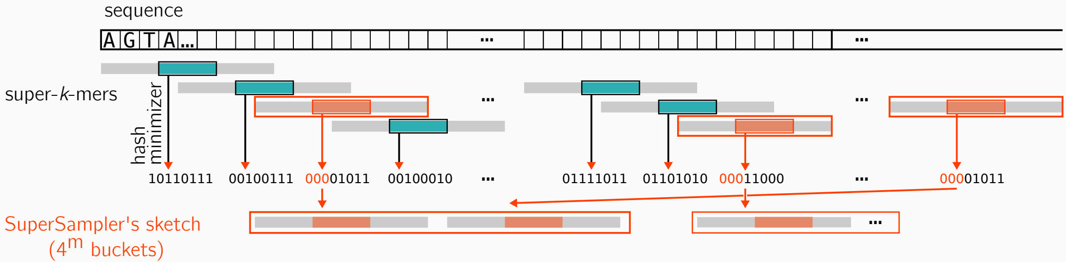


Comparison with experimental results

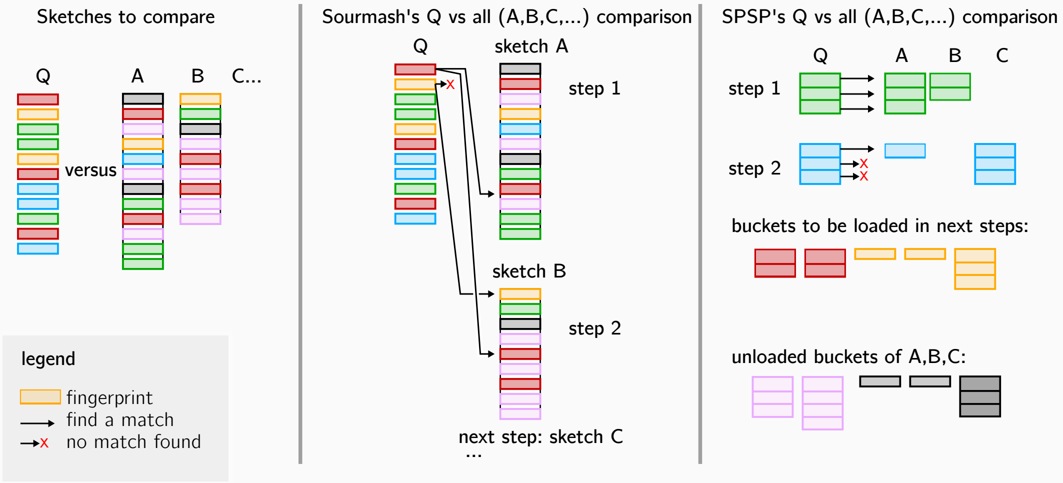


Methods

SuperSampler's sketches



Sketch comparison

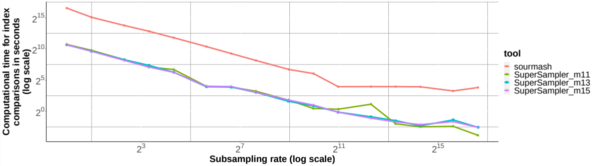


Results

Performance comparison on dissimilar data (RefSeq)

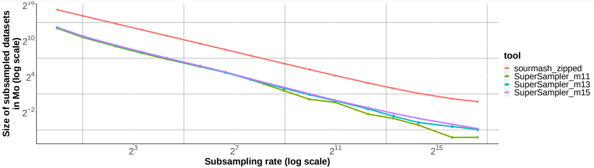
Computational time

50× faster



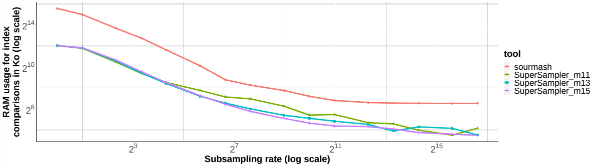
Disk usage

20× lighter



RAM usage

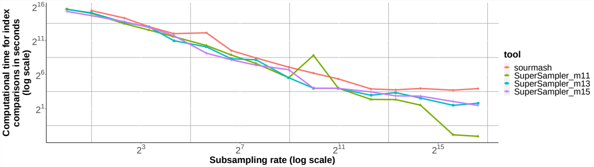
6× less RAM



Performance comparison on similar data (Salmonellas)

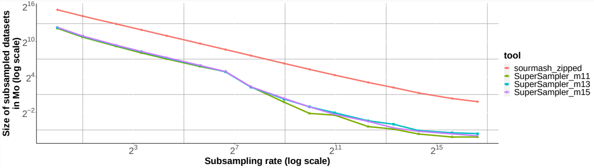
Computational time

7× faster



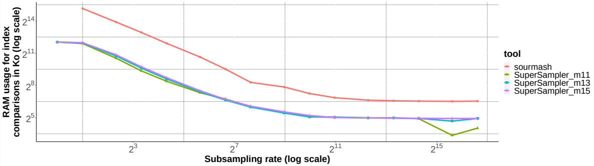
Disk usage

50× lighter



RAM usage

6× less RAM



Conclusion

Take home messages

- Super-k-mers
  - robust fingerprints
  - low memory cost
- Fractional Hitting Sets
  - generalization of UHS
  - lower density
  - longer super-k-mers
  - can be combined w/ existing UHS

- Go check out our posters!
  - **Igor Martayan's poster (#123)** on the theory behind FHS
  - **My poster about SuperSampler (#147)**, if you want to have a chat with me !
- If you want to dig deeper into FHS and SuperSampler, go checkout our preprint:

Appendix

References I

Brian D Ondov, Todd J Treangen, Páll Melsted, Adam B Mallonee, Nicholas H Bergman, Sergey Koren, and Adam M Phillippy.
Mash: fast genome and metagenome distance estimation using minhash.
Genome biology, 17(1):1–14, 2016.

N Tessa Pierce, Luiz Irber, Taylor Reiter, Phillip Brooks, and C Titus Brown.
Large-scale sequence comparisons with sourmash.
F1000Research, 8, 2019.

Mahmudur Rahman Hera, N Tessa Pierce-Ward, and David Koslicki.
Debiasing fracminhash and deriving confidence intervals for mutation rates across a wide range of evolutionary distances.
bioRxiv, 2022.

Kenneth Katz, Oleg Shutov, Richard Lapoint, Michael Kimelman, J Rodney Brister, and Christopher O'Sullivan.
The sequence read archive: a decade more of explosive growth.
Nucleic acids research, 50(D1):D387–D390, 2022.

Andrei Z Broder.
On the resemblance and containment of documents.
In *Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No. 97TB100171)*, pages 21–29. IEEE, 1997.

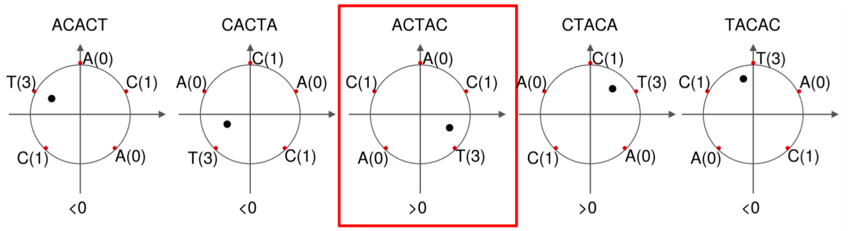
Hongyu Zheng, Carl Kingsford, and Guillaume Marçais.
Improved design and analysis of practical minimizers.
Bioinformatics, 36(Supplement_1):i119–i127, 2020.

David Pellow, Lianrong Pu, Baris Ekim, Lior Kotlar, Bonnie Berger, Ron Shamir, and Yaron Orenstein.
Efficient minimizer orders for large values of k using minimum decycling sets.
bioRxiv, pages 2022–10, 2022.

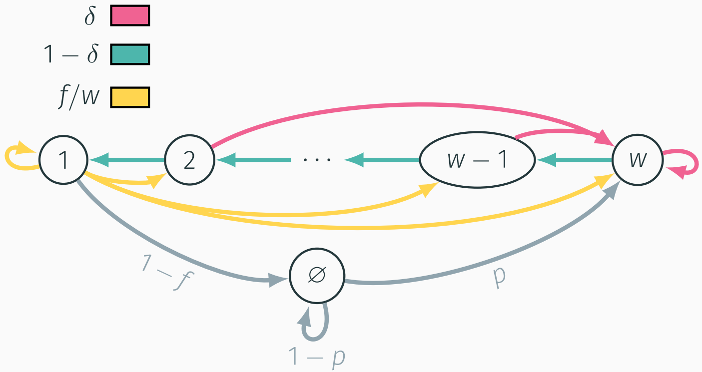
Decycling set

set S of m-mers whose removal make the De Bruijn graph acyclic

- if at least one m-mer is in S, take it in your UHS
- otherwise, use a random order to select a minimizer



Super-*k*-mers' Markov chain



- state i: small minimizer starts at i in the k-mer
- state $\varnothing$: no small minimizer in the k-mer