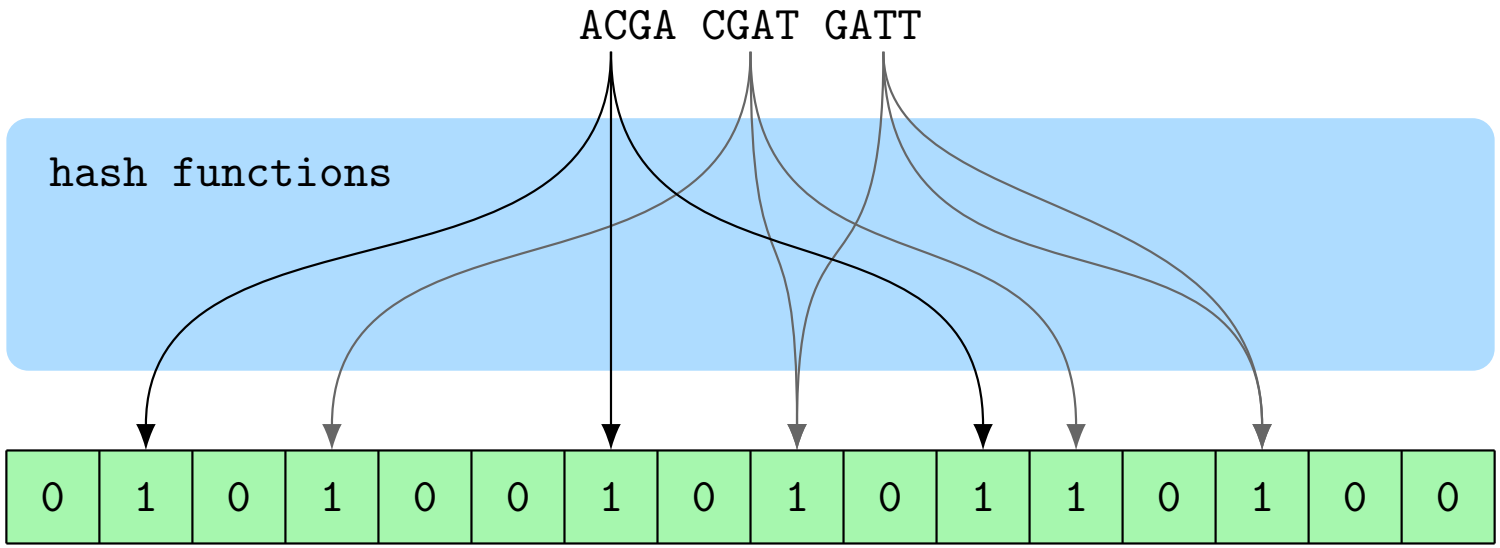
Karlsruhe Institute of Technology

COBS: A Compact Bit-Sliced Signature Index

Timo Bingmann, Phelim Bradley, Florian Gauger, and Zamin Iqbal · 2020-01-20

Extended Algorithmen II Edition · Originally @ SPIRE'19

INSTITUTE OF THEORETICAL INFORMATICS – ALGORITHMICS



Abstract

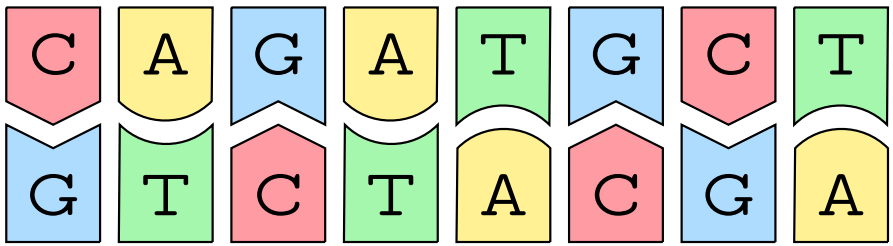
We present COBS, a COmpact Bit-sliced Signature index, which is a cross-over between an inverted index and Bloom filters. Our target application is to index k-mers of DNA samples or q-grams from text documents and process approximate pattern matching queries on the corpus with a user-chosen coverage threshold. Query results may contain a number of false positives which decreases exponentially with the query length. We compare COBS to seven other index software packages on 100 000 microbial DNA samples. COBS' compact but simple data structure outperforms the other indexes in construction time and query performance with Mantis by Pandey et al. in second place. However, unlike Mantis and other previous work, COBS does not need the complete index in RAM and is thus designed to scale to larger document sets.

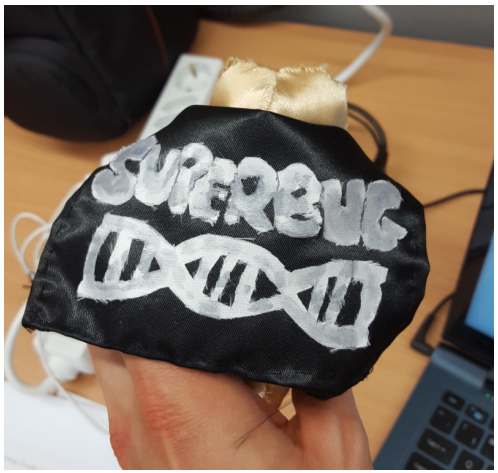
Motivation

Need approximate search in petabytes of DNA data.

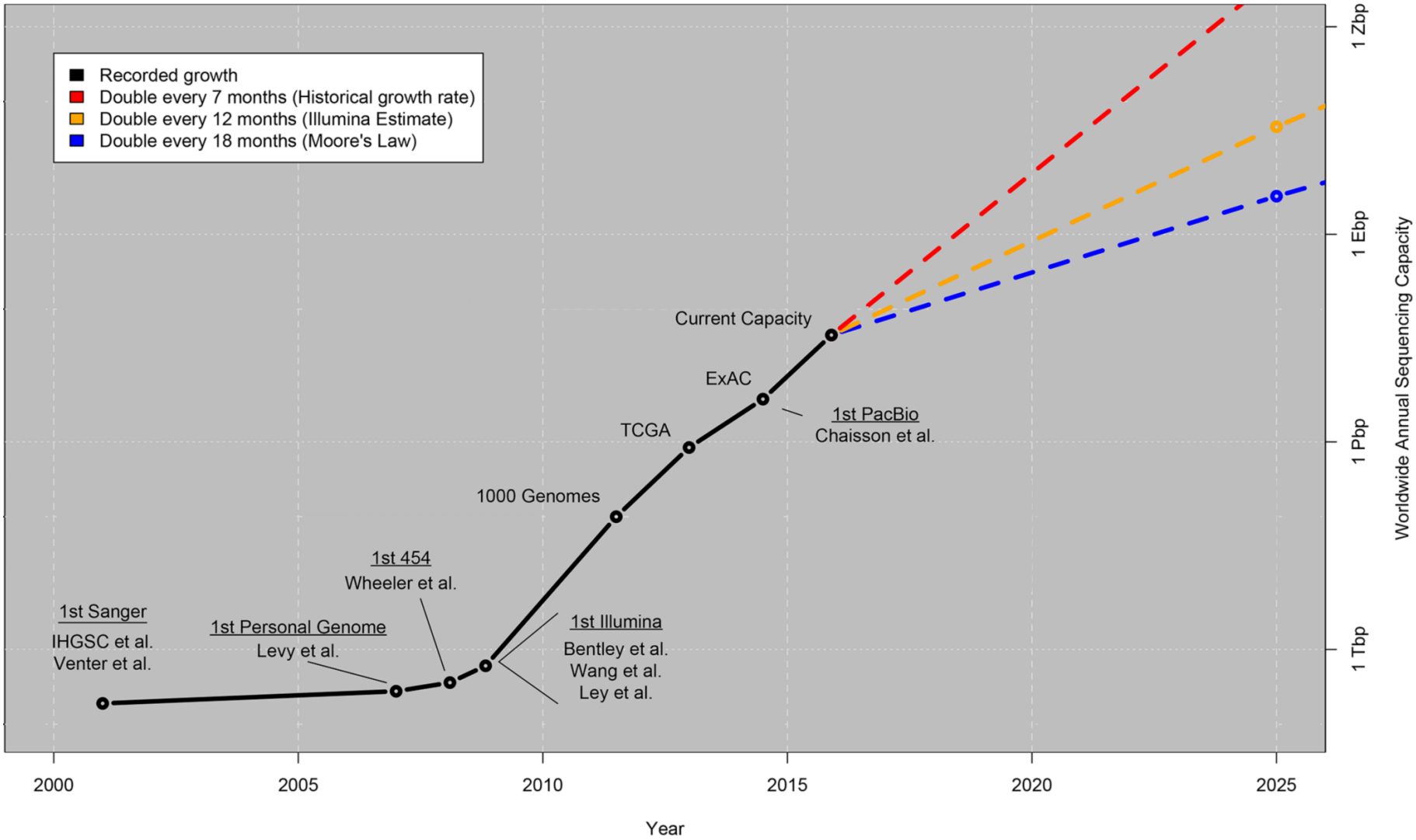
Applications:

- study global threats to public health
- epidemiology
- basic science of disease





Motivation



from Stephens et al. “Big data: astronomical or genomical?” (2015)

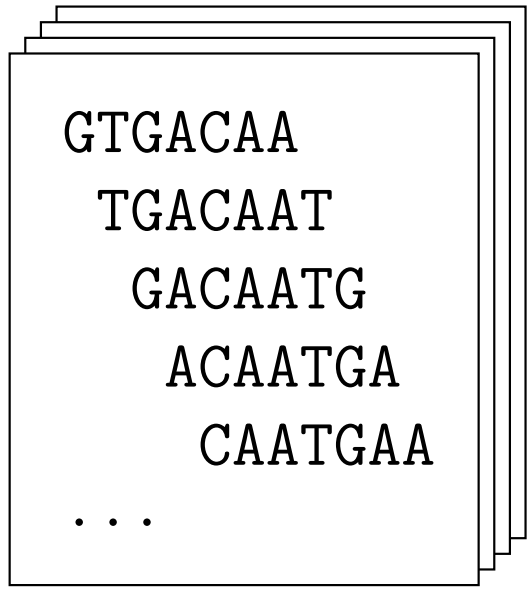
Approximate Pattern Matching

query

ATGACAATGACG

100-1000

documents

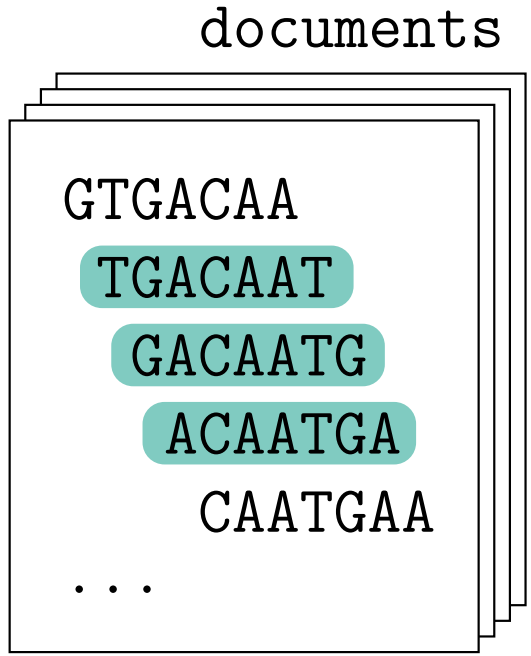


k-mers/q-grams

Approximate Pattern Matching

query
ATGACAATGACG

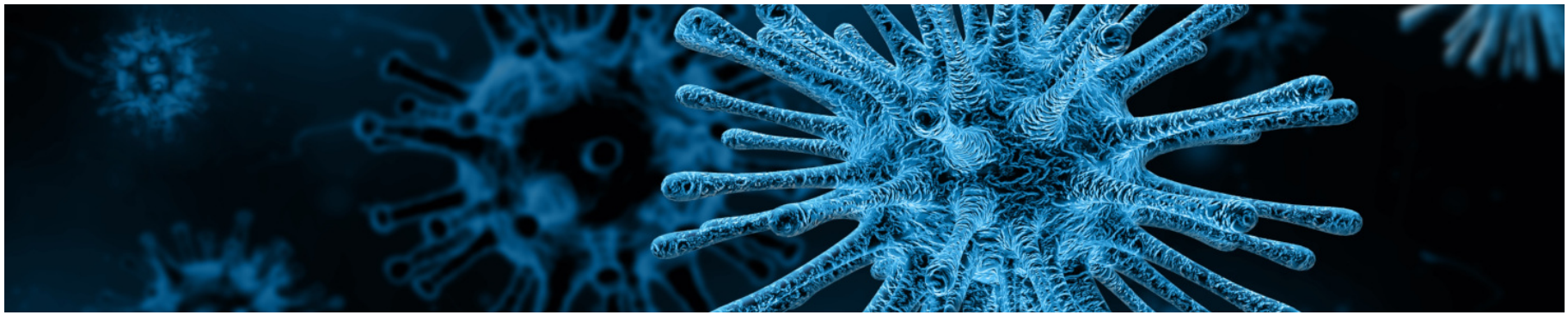
ATGACAA
TGACAAT
GACAATG
ACAATGA
CAATGAC
AATGACG



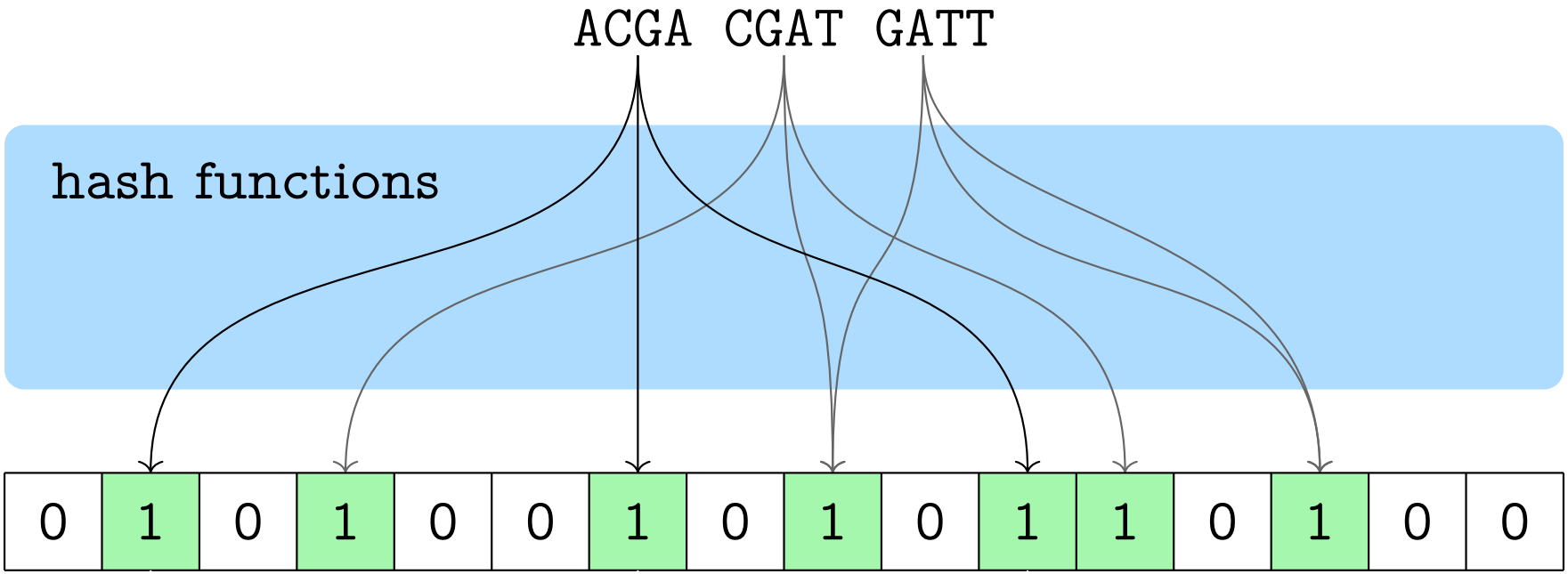
k-mers/q-grams

Related Work

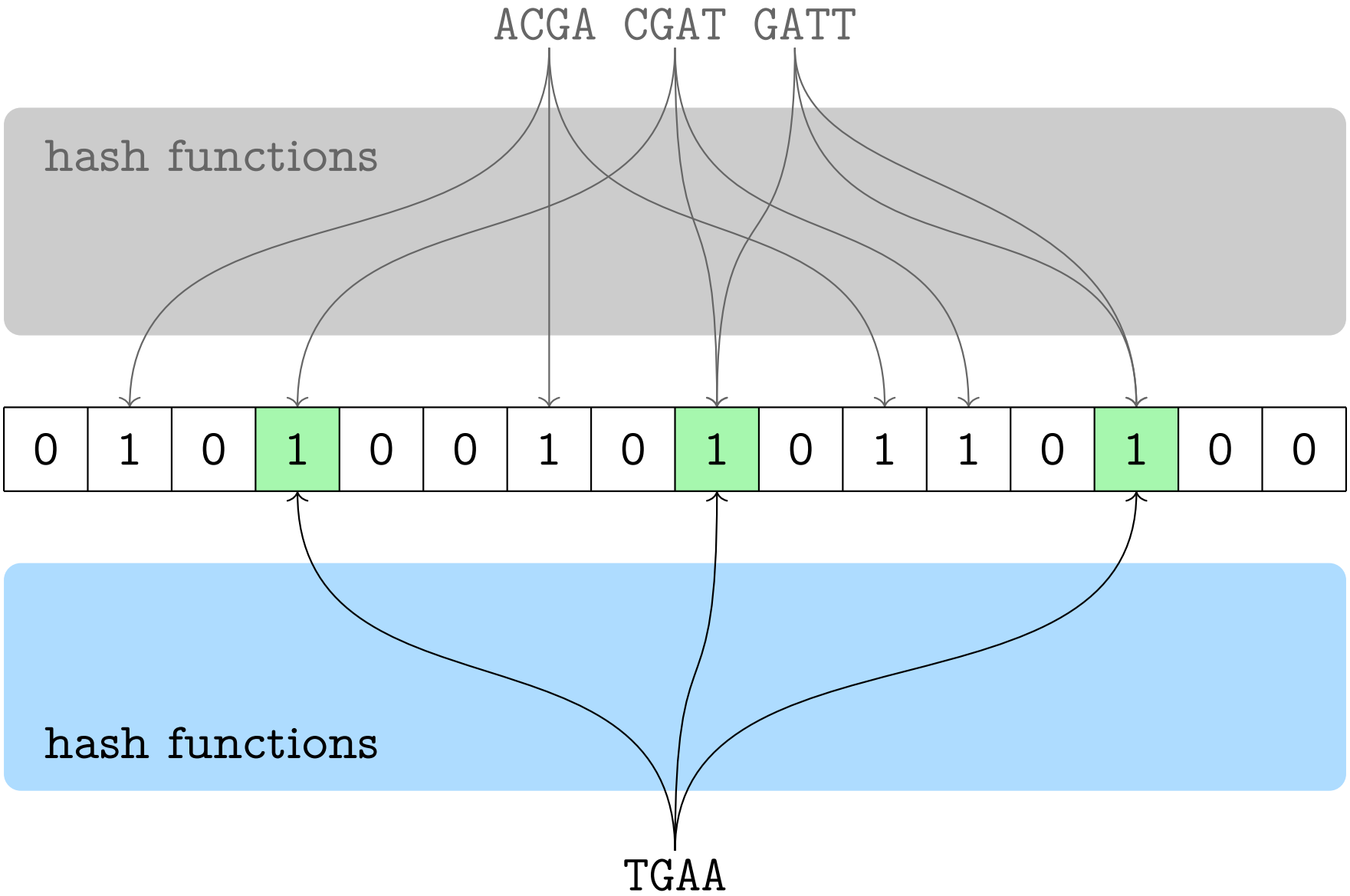
- Sequence Bloom Tree [SK16]
- Split Sequence Bloom Tree [SK18]
- AllSome Sequence Bloom Tree [Sun+18]
- HowDe Sequence Bloom Tree [HM18]
- SeqOthello [Yu+18]
- MANTIS [Pan+18]
- Bitsliced Genomic Signature Index [Bra+19]



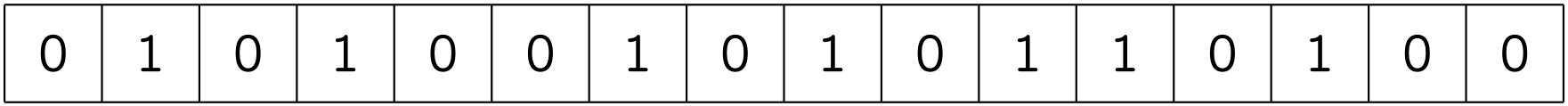
Bloom Filter



Bloom Filter



Bloom Filter Parameters



Parameters:

- m the array size (in bits)
- n items inserted into filter
- k mutually independent hash functions

Operations:

- *insert*(x) – Insert an item x.
- *query*(x) – Test if item x was inserted, with one-sided error. May return *true* despite x not having been inserted.
- Deletions are not easily supported.

Bloom Filter Parameters

0	1	0	1	0	0	1	0	1	0	1	1	0	1	0	0

Parameters:

- m the array size (in bits)
- n items inserted into filter
- k mutually independent hash functions

False Positive Probability:

- probability a bit is one: $1 - (1 - \frac{1}{m})^{kn}$.
- probability a membership query fails (false positive rate p_f):

$$p_f = \left(1 - \left(1 - \frac{1}{m}\right)^{kn}\right)^k \approx (1 - e^{-k\frac{n}{m}})^k .$$

Optimal Bloom Filter Parameters

0	1	0	1	0	0	1	0	1	0	1	1	0	1	0	0

Optimal Parameters:

- $k = \frac{m}{n} \ln 2$ hash functions given m bits and n items.
- $m = -n \cdot \frac{\ln p_f}{(\ln 2)^2}$ bits in filter given p_f and n items.

Optimal Bloom Filter Parameters

We have $p_f = (1 - e^{-k\frac{n}{m}})^k = \exp(k \ln(1 - e^{-k\frac{n}{m}}))$.

To find the optimal number of hash functions k, determine a root of the partial derivative of $\ln(p_f)$ which locates a minimum of p_f:

$$\frac{\partial \ln(p_f)}{\partial k} = \ln(1 - e^{-k\frac{n}{m}}) + \frac{k}{1 - e^{-k\frac{n}{m}}} \frac{\partial(1 - e^{-k\frac{n}{m}})}{\partial k}$$

$$= \ln(1 - e^{-k\frac{n}{m}}) + \frac{k}{1 - e^{-k\frac{n}{m}}} \cdot \frac{n}{m} \cdot e^{-k\frac{n}{m}} .$$

Trying $k = c \cdot \frac{m}{n}$ as solution yields

$$0 = \ln(1 - e^{-c}) + \frac{c}{1 - e^{-c}} e^{-c} .$$

Further assuming $c = \ln x$ yields

$$0 = \ln\left(1 - \frac{1}{x}\right) + \frac{\ln x}{1 - \frac{1}{x}} \frac{1}{x} = \ln\left(1 - \frac{1}{x}\right) + \frac{\ln x}{x - 1}$$

for which $x = 2$ is a solution.

Optimal Bloom Filter Parameters

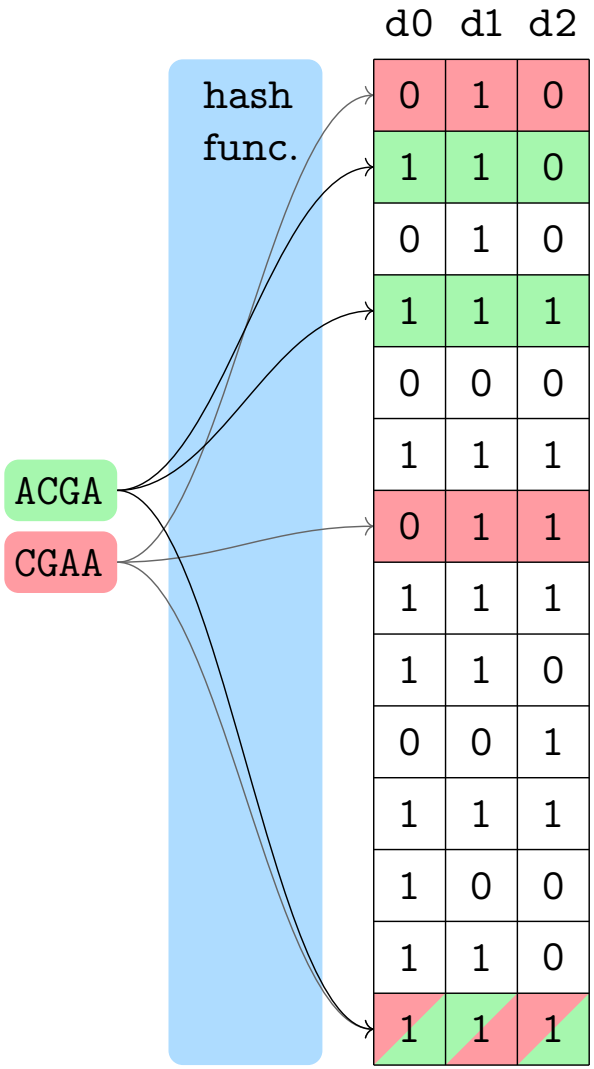
We can insert $k = \frac{m}{n} \ln 2$ into the equation for p_f which determines the optimal false positive rate given a fill $\frac{m}{n}$ as

$$p_f = \left(1 - e^{-\frac{m}{n}(\log 2)\frac{n}{m}}\right)^{\frac{m}{n} \ln 2} \qquad \text{or} \qquad \ln(p_f) = -\frac{m}{n}(\ln 2)^2 .$$

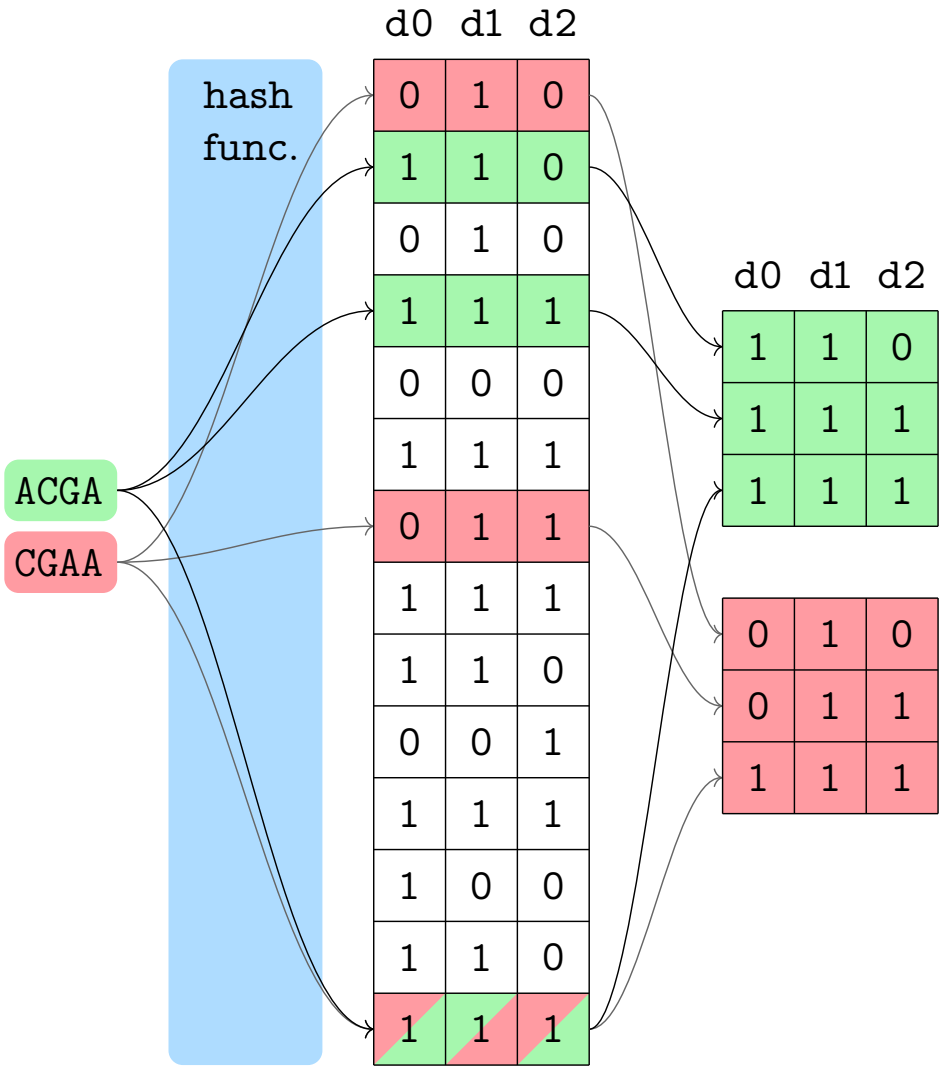
Hence, if the number of items n and desired false positive rate p_f are given, we have

$$m = -n \cdot \frac{\ln p_f}{(\ln 2)^2} .$$

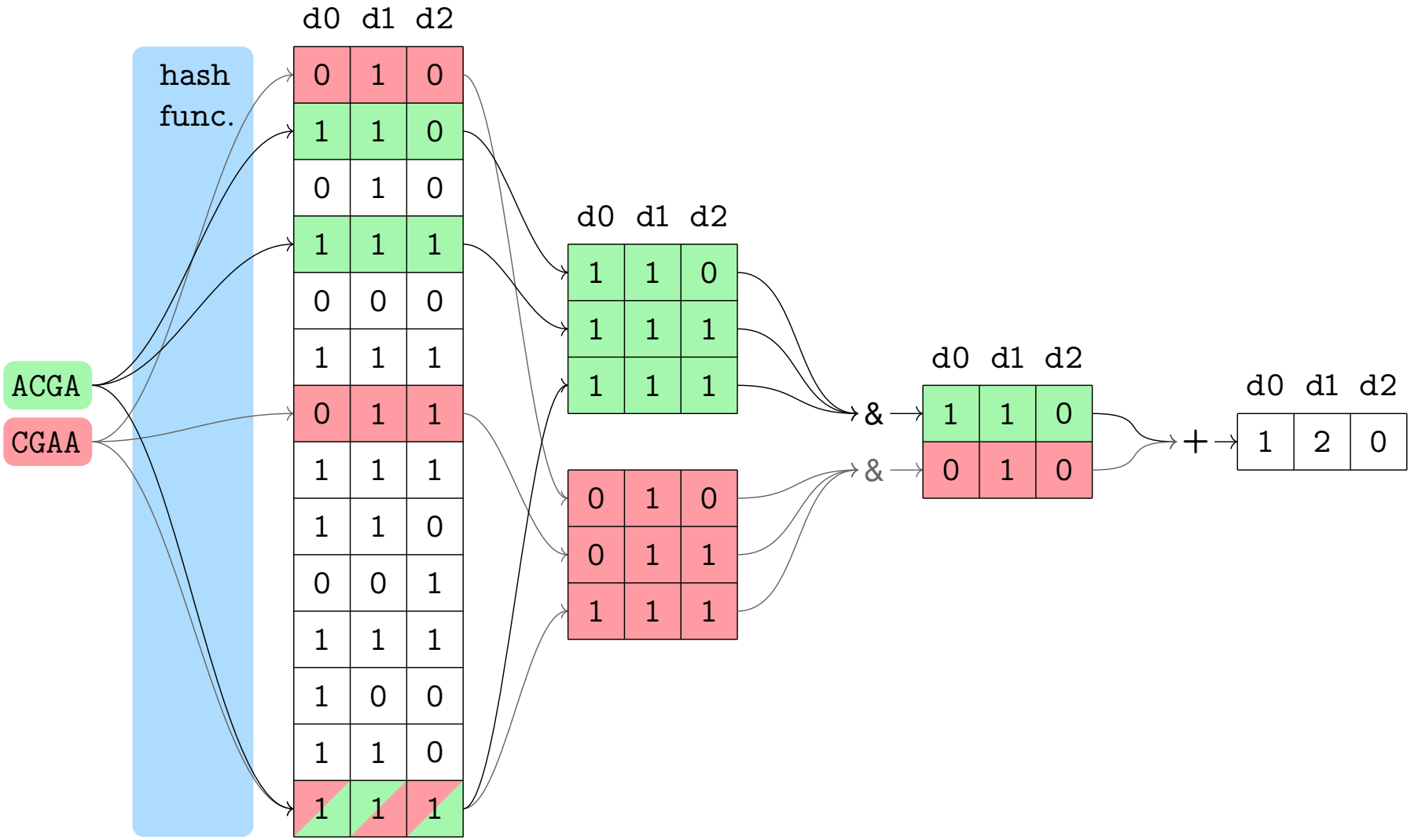
Bit-Sliced Signature Index



Bit-Sliced Signature Index



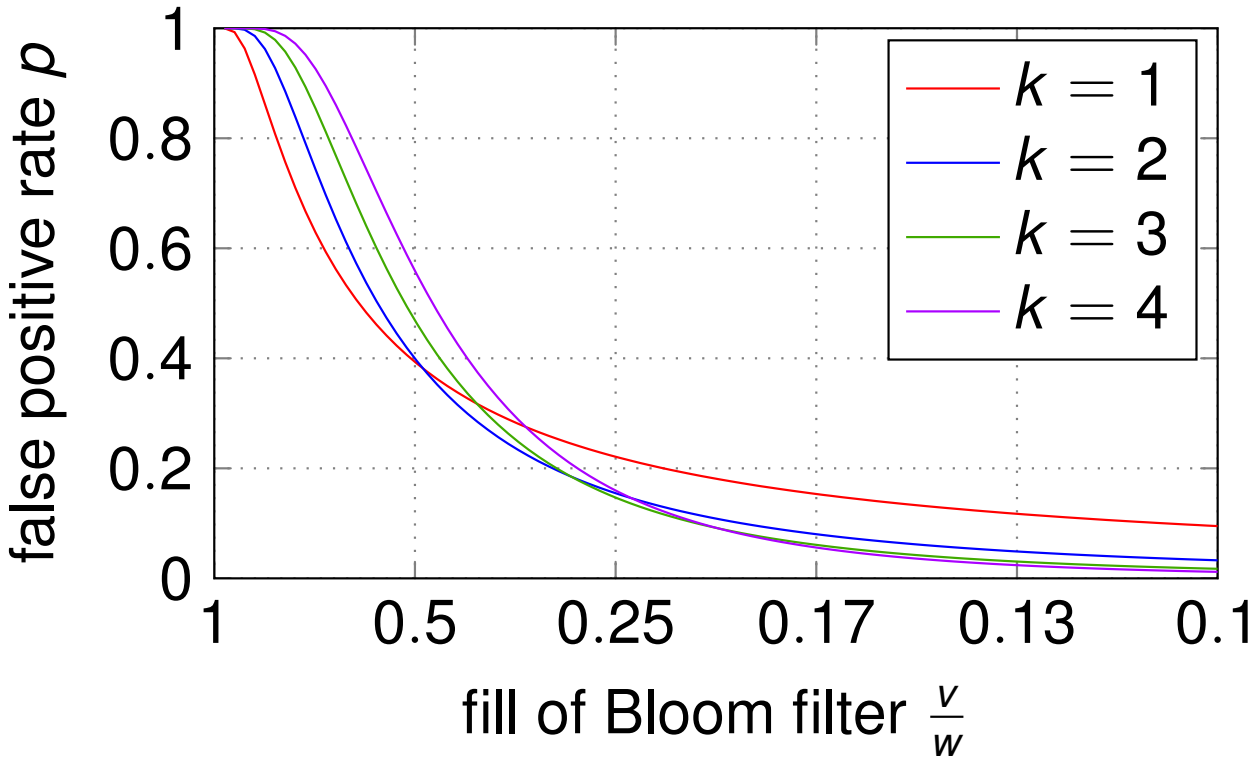
Bit-Sliced Signature Index



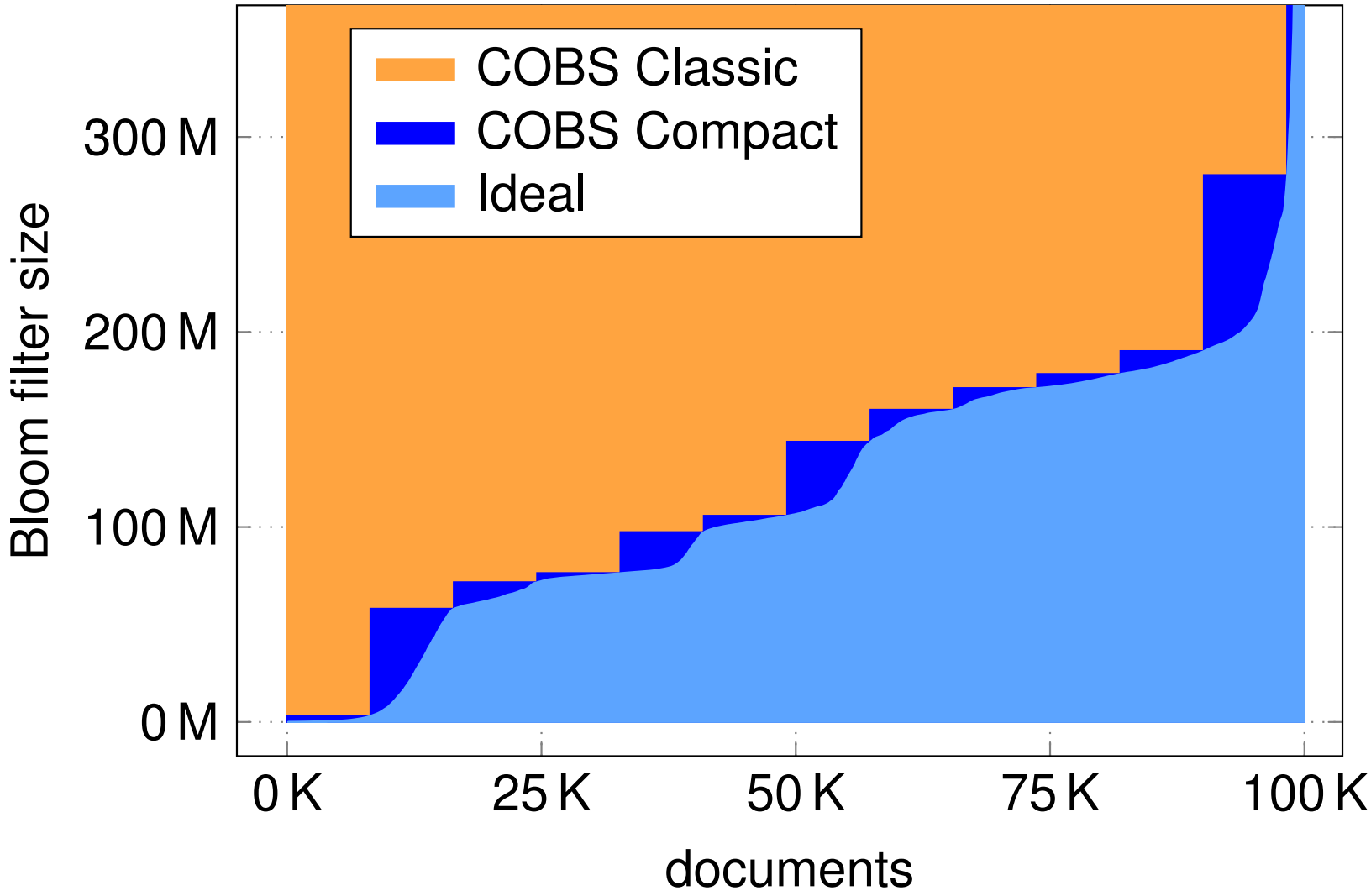
Bloom Filter Parameters

Theorem: False Positive Rate of a Query, Thm 2 in [SK16]

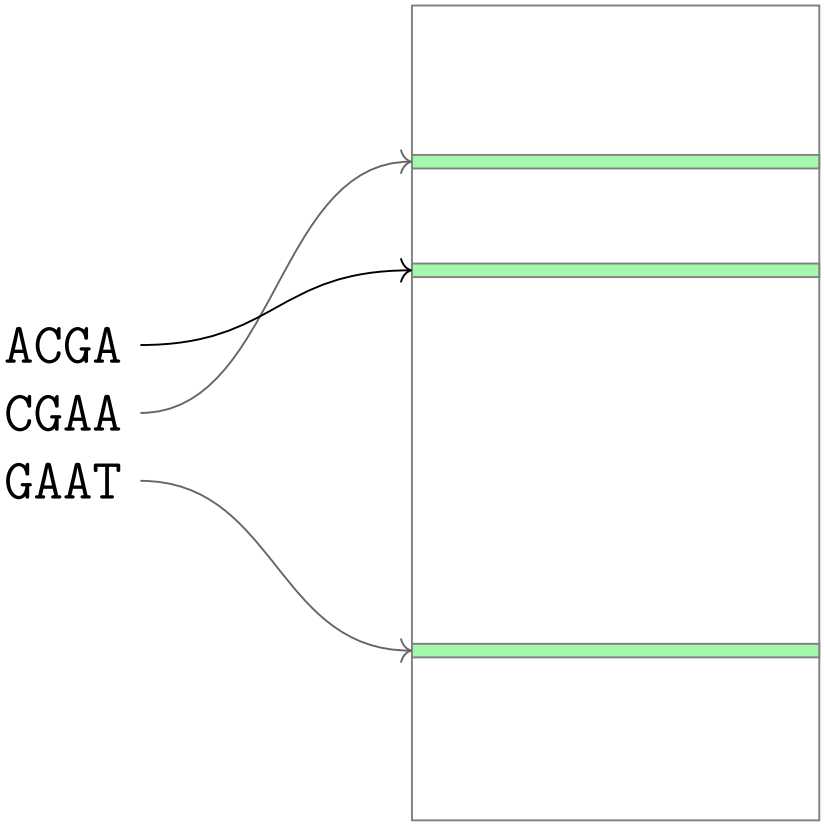
Let P be a query containing ℓ distinct q-grams and K a threshold. If we consider the terms as being independent, the probability that more than $\lfloor K\ell \rfloor$ false-positive terms occur in a filter f with false positive rate p is $1 - \sum_{i=0}^{\lfloor K\ell \rfloor} \binom{\ell}{i} p^i (1-p)^{\ell - i}$.



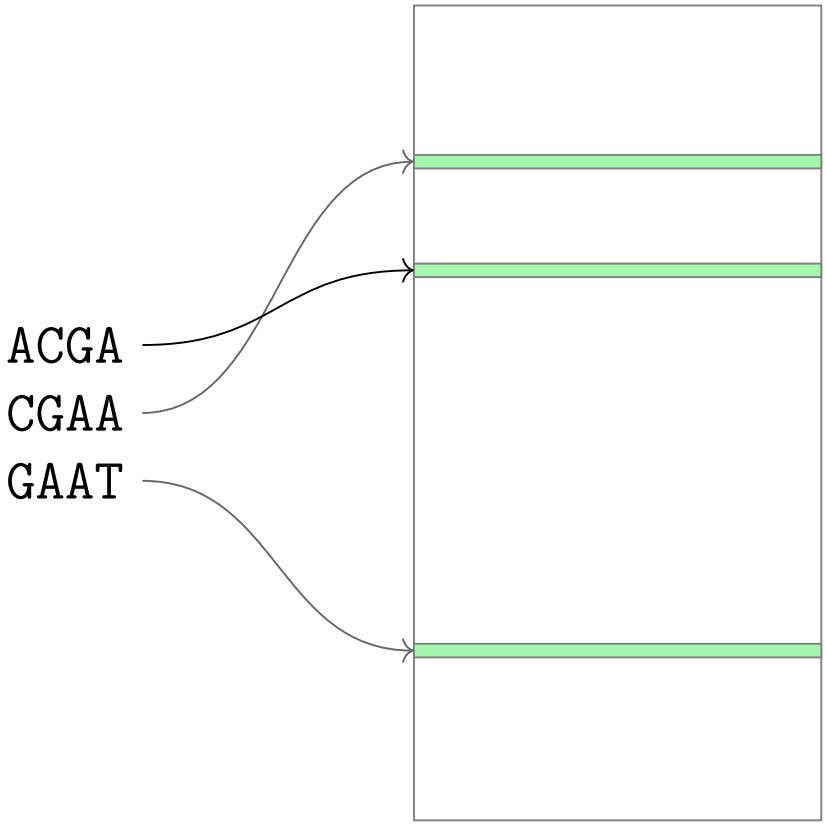
Compact Bit-Sliced Signature Index

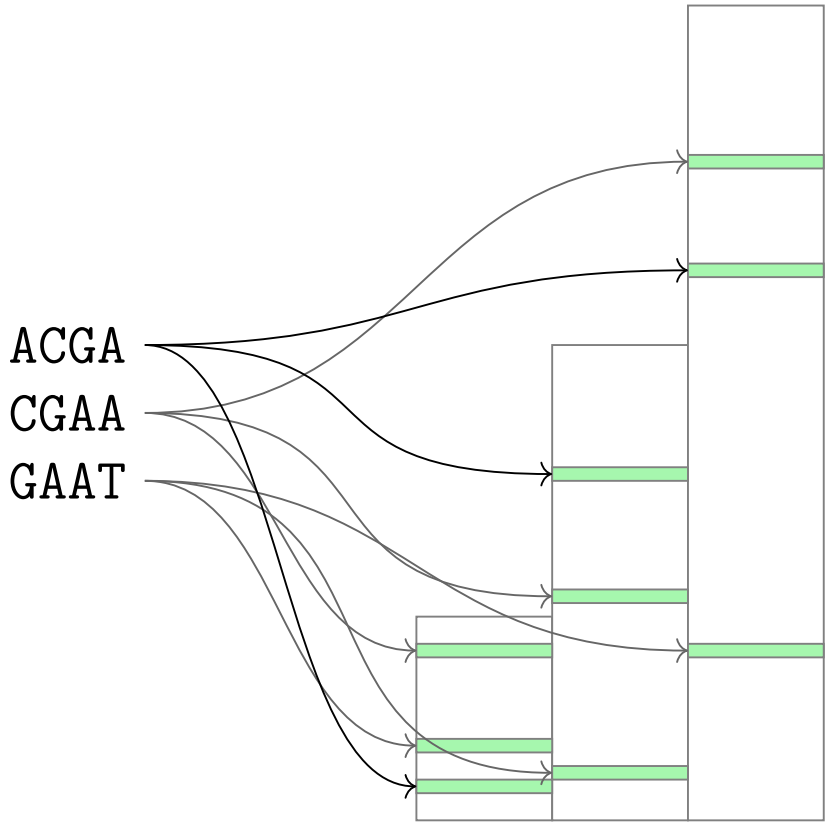


COBS: Disk Access Pattern



COBS: Disk Access Pattern





more about disk, SSD, and NVMe access pattern speeds:
`https://panthema.net/2019/0322-nvme-batched-block-access-speed/`

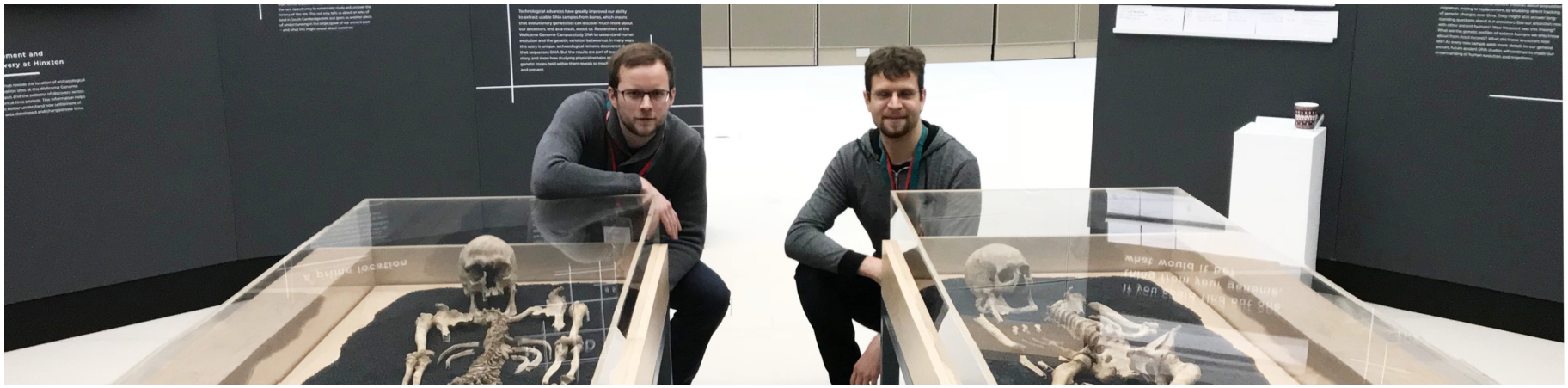
COBS: Summary

COBS Index Design: (values used in practice)

- use $k = 1$ hash functions with $f = 0.3$ false positive rate
- compact $\Theta(B) = 4\,\text{Ki}$ documents into subindices

COBS Software:

- C++ implementation started by Florian Gauger
- can read Text, Fasta, Fastq, and McCortex files
- parallelized and multi-level if needed construction
- SIMD instructions in query processing



Experiments – Software and Machine

Karlsruhe Institute of Technology

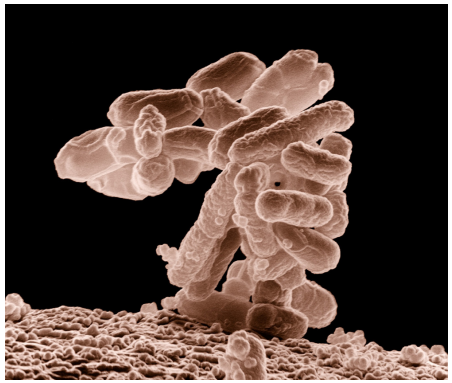
Eight Software Packages:

- Sequence Bloom Tree (SBT) [SK16]
- Split Sequence Bloom Tree (SSBT) [SK18]
- AllSome Sequence Bloom Tree (AllSome-SBT) [Sun+18]
- HowDe Sequence Bloom Tree (HowDe-SBT) [HM18]
- MANTIS [Pan+18]
- SeqOthello [Yu+18]
- Bitsliced Genomic Signature Index (BIGSI) [Bra+19]
- our Classic Bit-Sliced Index (Classic BSI) [this]
- and COBS [this]

Machine:

- Intel Gold 6138 2.0 GHz 4×20 cores with 768 GiB RAM.
- 4×2 TB NVMe Samsung 970 EVO SSD as software RAID 0.

Experiments – Data



Microbial Data:

- 100 000 microbial (viri and bacteria) documents from European Nucleotide Archive (ENA)
- Split into 100, 250, 500, 1 000, 2 500, . . ., 100 000 subsets.
- Average document size ≈ 42.77 MiB, ≈ 4 TiB in total.
- ENA contained $1.5 \cdot 10^9$ documents in 2018.

Queries: four batches, with

- length $\ell \in \{31,\ 100,\ 1\,000,\ 10\,000\}$, containing $q \in \{100\,000,\ 100\,000,\ 10\,000,\ 1\,000\}$ random true positives and q true negatives.
- Check each index software's results.

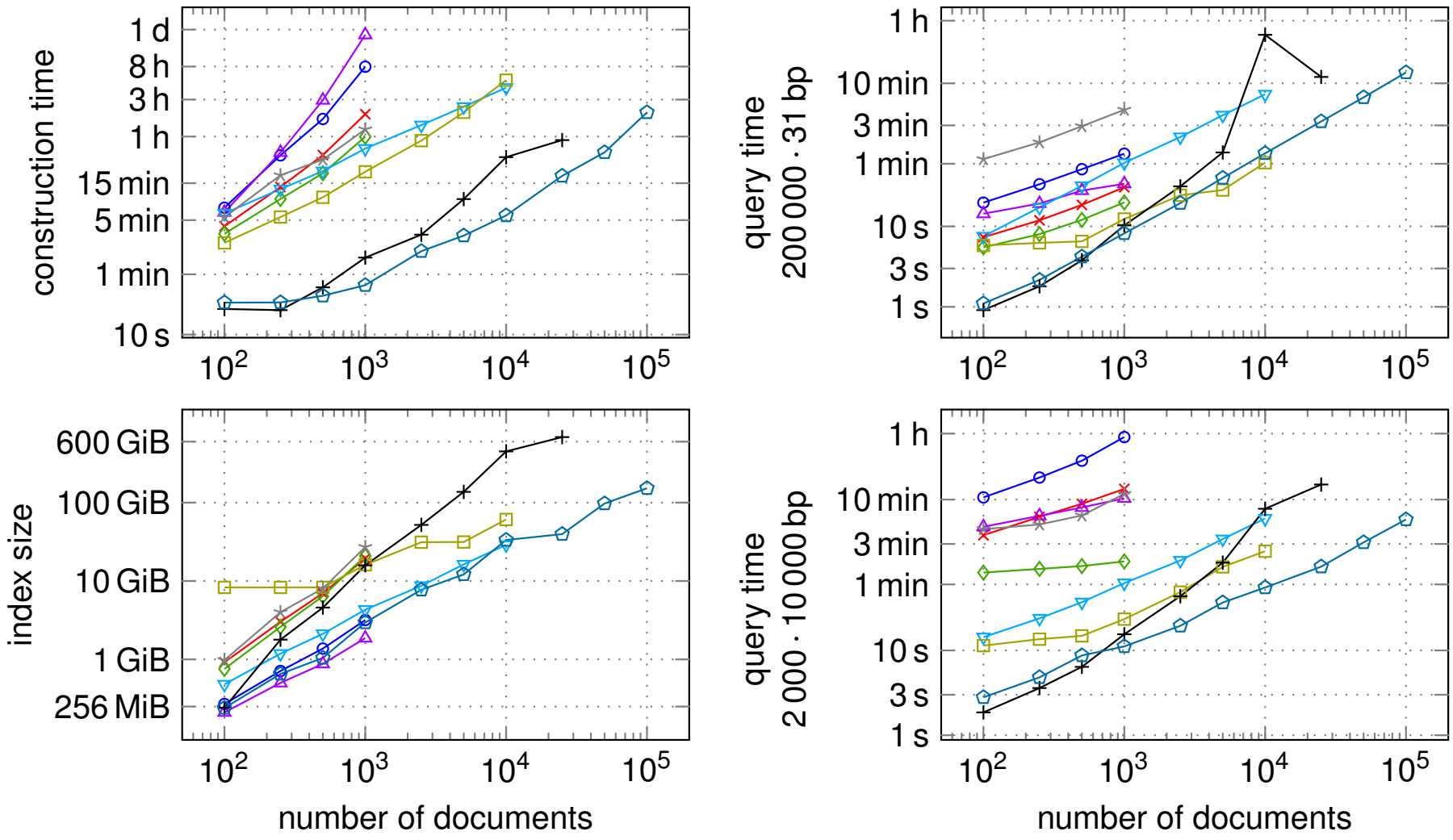
Results for 1000 Microbial Documents

phase	SBT	SSBT	AllSome-SBT	HowDe-SBT	Seq-Othello	Mantis	BIGSI	Classic BSI	COBS Compact
	Construction Wall-Clock Time in Seconds								
count	2 018	1 974	1 954	1 959					
bloom	114	117	140	144	295	232	1 881		
build	3 097	21 378	1 401	68 034	2 225	987	2 574	99	43
compress	1 768	5 187	80	3 802		45			
total	6 996	28 657	3 576	73 939	2 520	1 264	4 455	99	43
	Construction CPU (User) Time in Seconds								
count	4 574	4 511	4 475	4 488					
bloom	11 133	10 967	10 234	10 278	28 123	19 162	169 345		
build	855	5 178	449	66 872	2 198	943	1 767	1 604	1 430
compress	1 569	4 832	1 663	2 857		3 423			
total	18 131	25 489	16 821	84 495	30 320	23 527	171 113	1 604	1 430
	Construction Maximum RSS Memory Usage in MiB								
count	518	518	518	518					
bloom	641	640	640	640	634	1 756	4 244		
build	11 028	1 523	7 140	108 147	12 137	88 357	246 806	16 245	2 616
compress	10 953	992	560	963		16 613			
maximum	11 028	1 523	7 140	108 147	12 137	88 357	246 806	16 245	2 616
	Index Size in MiB								
size	19 844	3 254	21 335	1 911	4 410	16 486	27 794	16 236	3 022

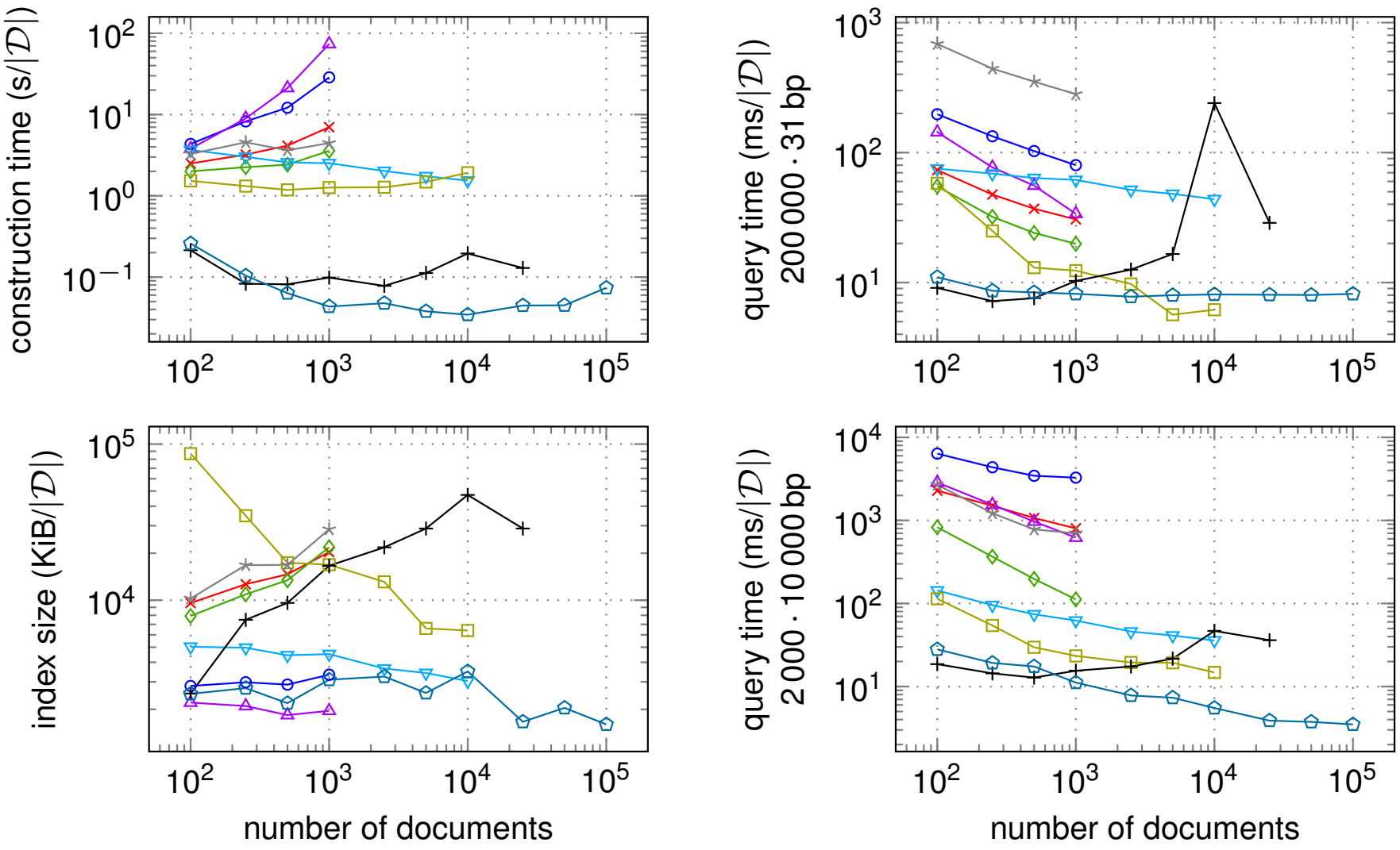
Results for 1000 Microbial Documents

phase	SBT	SSBT	AllSome-SBT	HowDe-SBT	Seq-Othello	Mantis	BIGSI	Classic BSI	COBS Compact
ℓ	Query Wall-Clock Time in Seconds								
31 bp r0	31	80	20	34	62	12	281	10	8
31 bp r2	26	76	19	33	62	13	289	9	8
100 bp r0	663	3 183	100	600	73	22	783	14	9
100 bp r2	649	3 153	95	588	73	23	455	14	9
1000 bp r0	794	3 466	112	670	63	21	660	15	10
1000 bp r2	781	3 435	108	659	64	27	310	13	10
10000 bp r0	802	3 273	112	622	62	23	699	16	11
10000 bp r2	790	3 243	111	613	62	22	316	15	11
total r0–r2	6 775	29 833	1 007	5 710	783	252	5 177	154	114
	Document False Positive Rate for 31 bp Queries								
rate	0.004	0.004	0.004	0.004	0.001	0.000	0.027	0.024	0.227

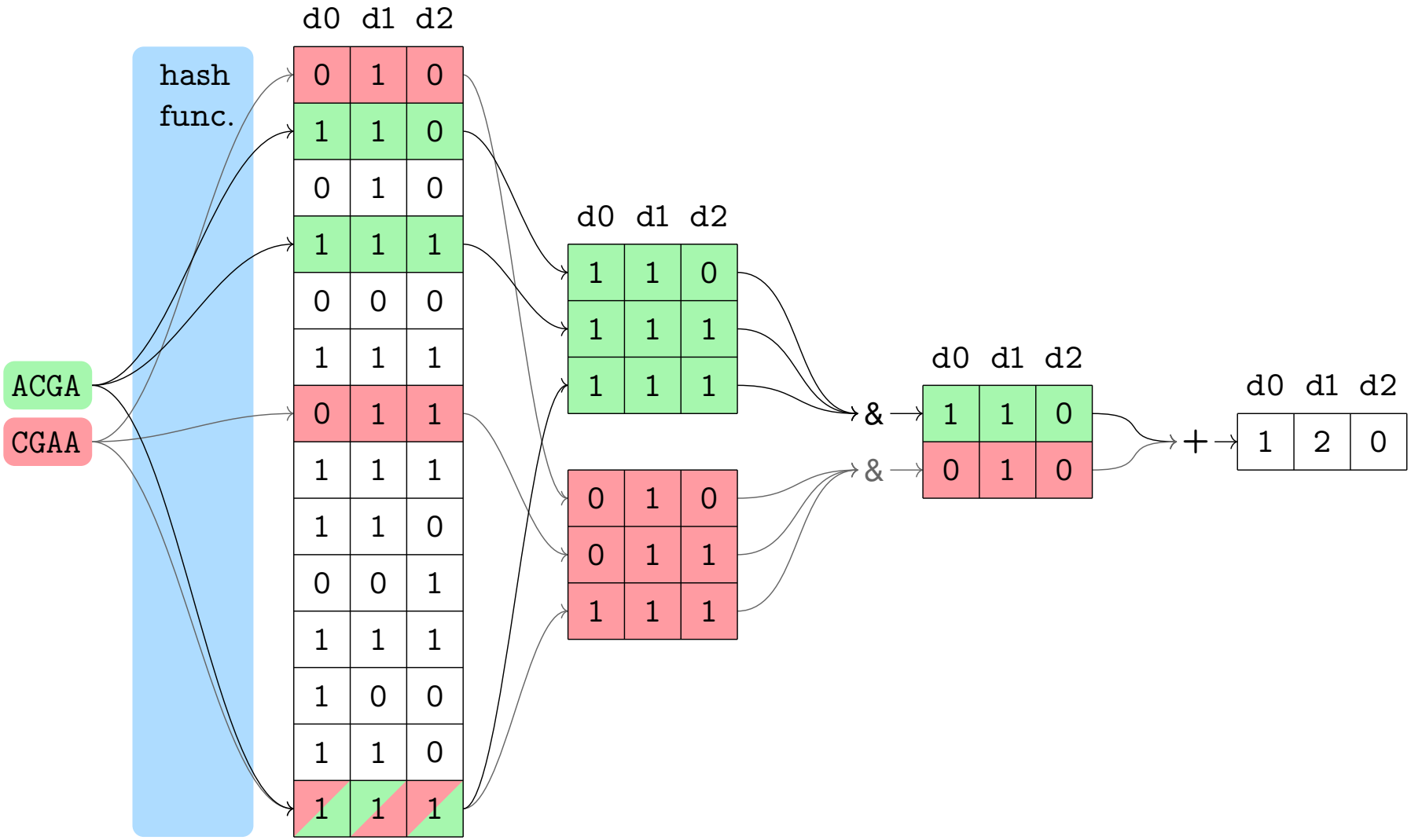
Scaling Results Microbial Documents



Scaling Results Microbial Documents



Bit-Sliced Signature Index



Conclusion

Karlsruhe Institute of Technology

Software:

- COBS is available as open source:
  `https://panthema.net/cobs/`
- soon: more documentation and Python front-end module

Future Work:

- Daniel Ferizovic tried clustering of documents
- also working on dealing with insertions and deletions
- batched query processing e.g. for whole genomes
- distributed COBS query processing for ENA-scale index
- adapt completely different filter for use as an index

Questions?