Compressing and indexing pangenomes with *meta-colored* compacted *de Bruijn* graphs

Giulio Ermanno Pibiri

Ca’ Foscari University of Venice and ISTI-CNR

@giulio_pibiri

@jermp

Joint work with
Jason Fan and **Rob Patro**
University of Maryland (USA)

YRA Seminar, 2 November 2023

The colored k-mer indexing problem

- We are given a collection $\mathscr{R} = \{R_1, \ldots, R_N\}$ of reference sequences.
  Each R_i is a (large) sequence over the DNA alphabet {A,C,G,T}.

- A **k-mer** is a sub-string of length k of some R_i.

- A lot of hype in the indexing community for the case where $\mathscr{R}$ is a *pangenome*, i.e., a collection of related genomes.

- **Problem.** We want to build an *index* for $\mathscr{R}$ so that we can retrieve the set Color$(x) = \{i \,|\, x \in R_i\}$ efficiently for any k-mer x. Note that Color$(x) = \emptyset$ if $x \notin \mathscr{R}$.

The colored k-mer indexing problem

- We are given a collection $\mathscr{R} = \{R_1, \ldots, R_N\}$ of reference sequences. Each R_i is a (large) sequence over the DNA alphabet {A,C,G,T}.

- A **k-mer** is a sub-string of length k of some R_i.

- A lot of hype in the indexing community for the case where $\mathscr{R}$ is a *pangenome*, i.e., a collection of related genomes.

- **Problem.** We want to build an *index* for $\mathscr{R}$ so that we can retrieve the set Color$(x) = \{i \,|\, x \in R_i\}$ efficiently for any k-mer x. Note that Color$(x) = \emptyset$ if $x \notin \mathscr{R}$.

- **Applications:** This problem is relevant for applications where sequences are first matched against known references (i.e., mapping/alignment algorithms): single-cell RNA-seq, metagenomics, etc.

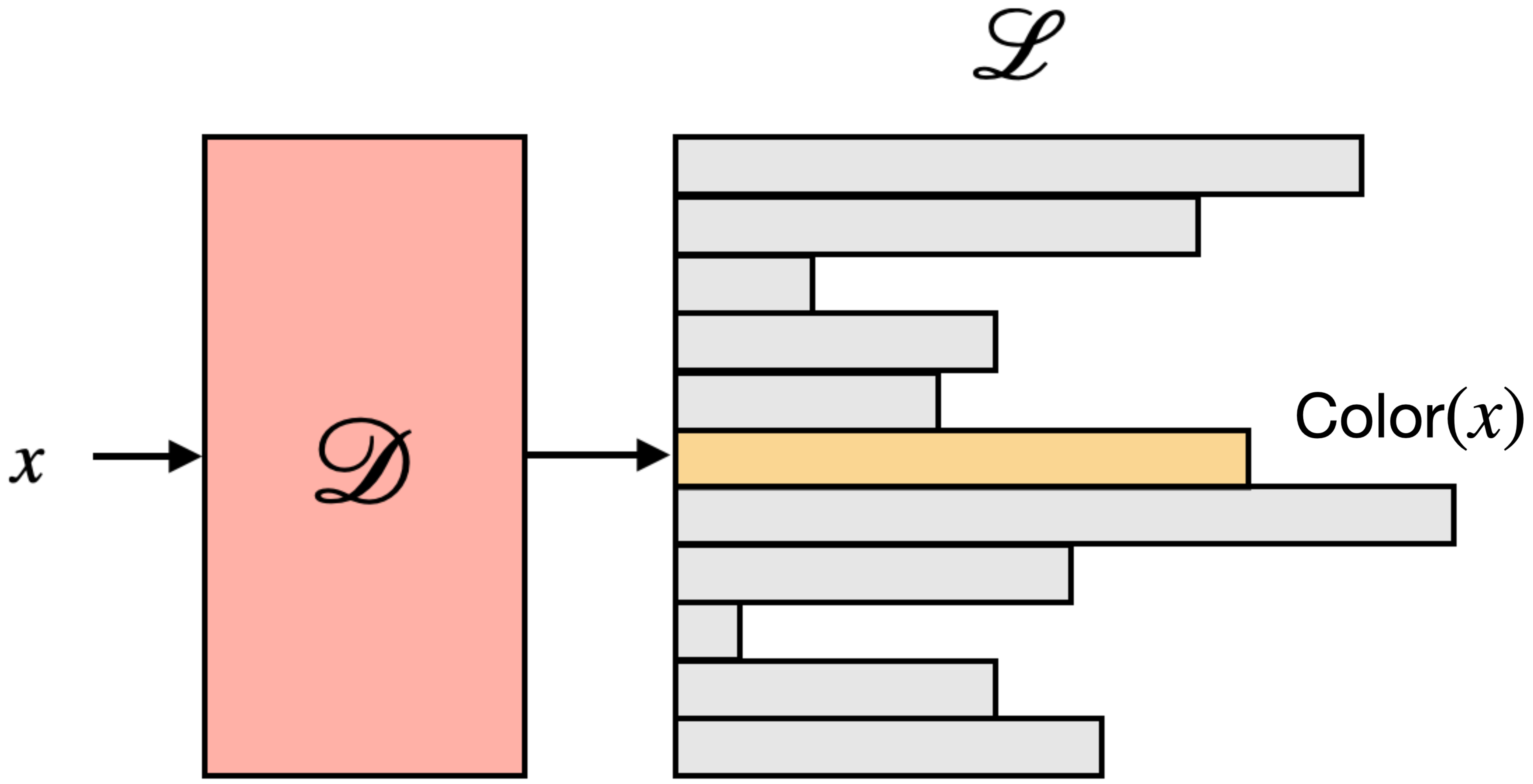
Modular indexing layout

- All the distinct k-mers in $\mathscr{R} = \{R_1, \ldots, R_N\}$ are stored in the **dictionary** $\mathscr{D}$.

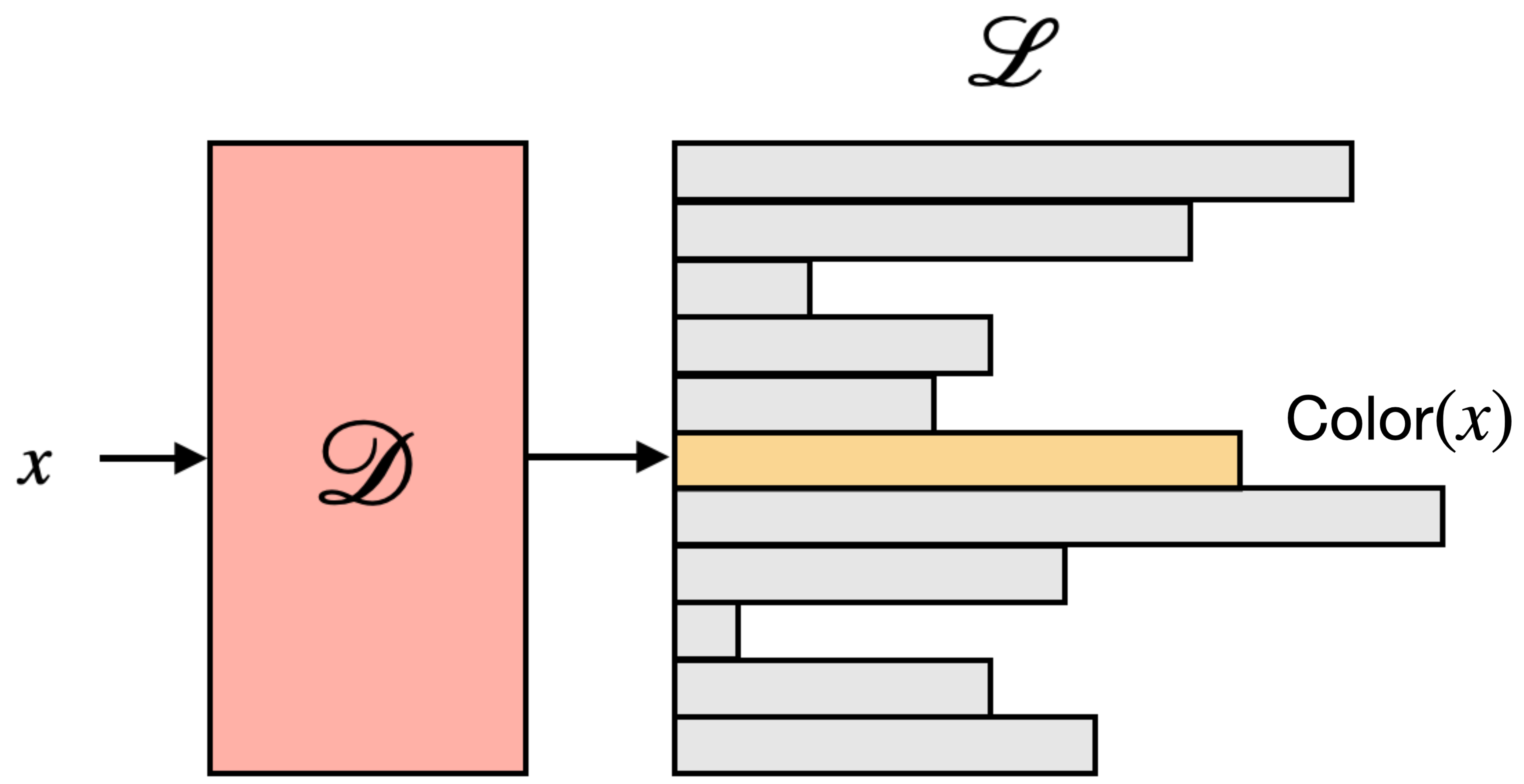
- What we want for a k-mer x is the map:

$$x \rightarrow \text{Color}(x) = \{i \mid x \in R_i\}.$$

The collection of all Color(x) is the **inverted index** $\mathscr{L}$.



Modular indexing layout



- All the distinct k-mers in $\mathscr{R} = \{R_1, \ldots, R_N\}$ are stored in the **dictionary** $\mathscr{D}$.

- What we want for a k-mer x is the map:

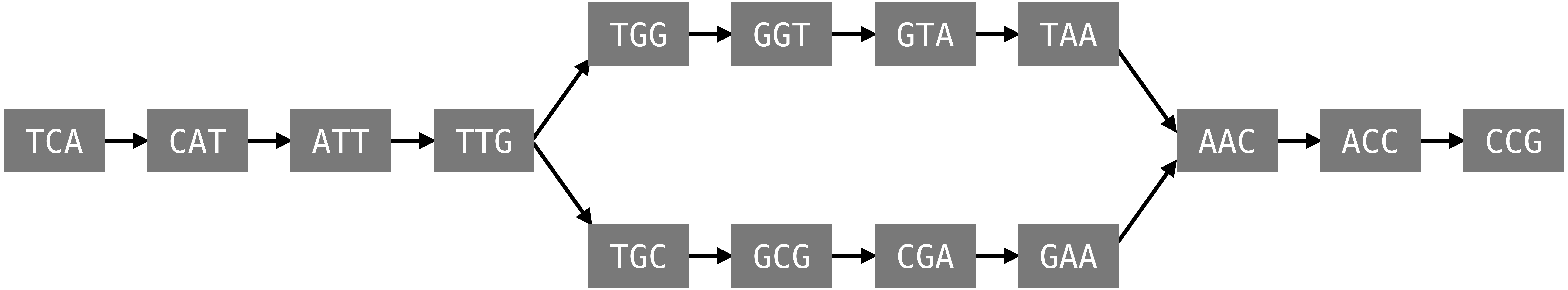
  $x \rightarrow \text{Color}(x) = \{i \,|\, x \in R_i\}$.

  The collection of all Color(x) is the **inverted index** $\mathscr{L}$.

- Our problem reduces to that of **representing two data structures**, $\mathscr{D}$ and $\mathscr{L}$.

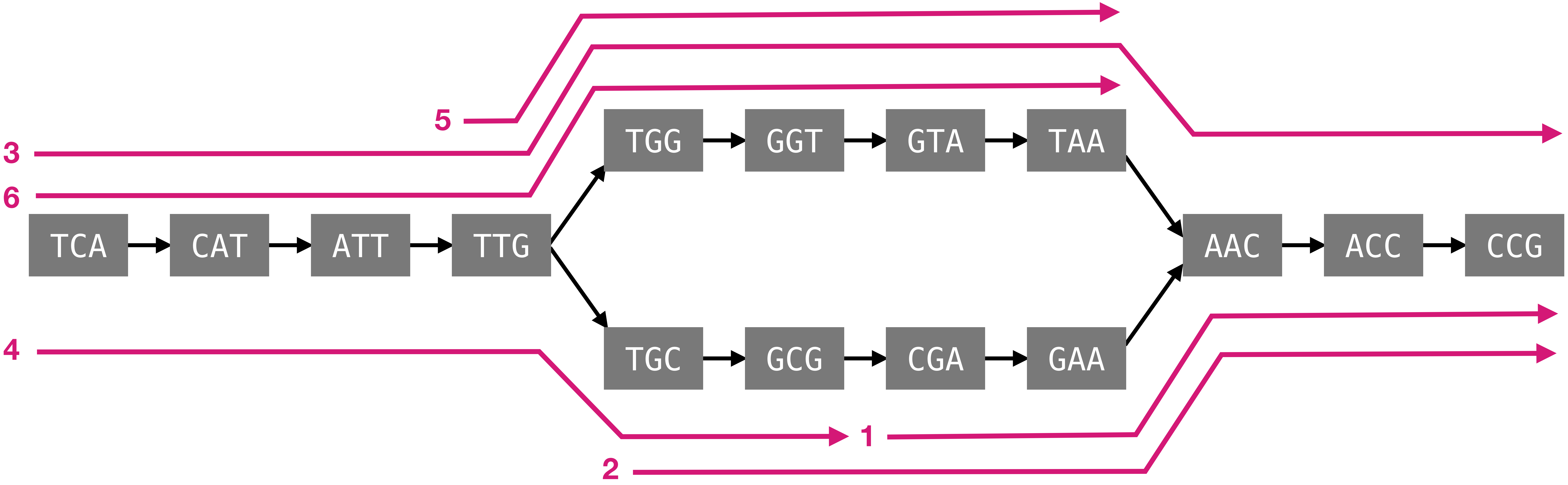
de Bruijn graphs

- One-to-one correspondence between a set of k-mers and a *de Bruijn* graph (dBG).
- Example for **k = 3**.



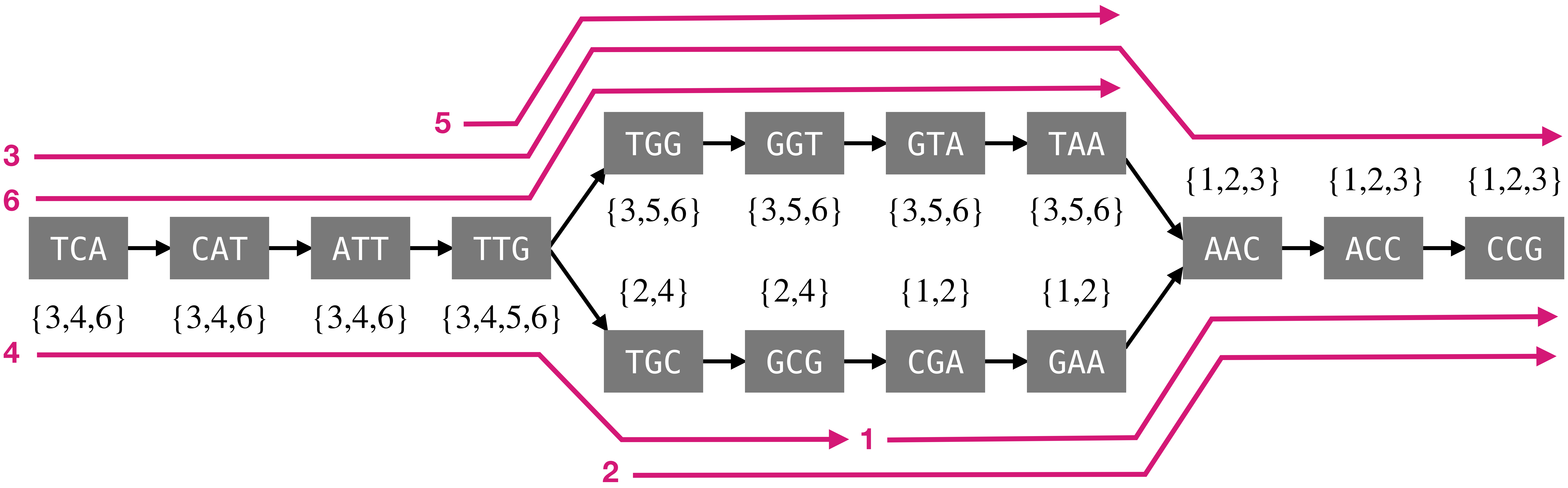
Colored *de Bruijn* graphs

- One-to-one correspondence between a set of k-mers and a *de Bruijn* graph (dBG).
- Example for k = 3 and **N = 6** references. References in $\mathscr{R}$ are spelled by **paths** in the graph.



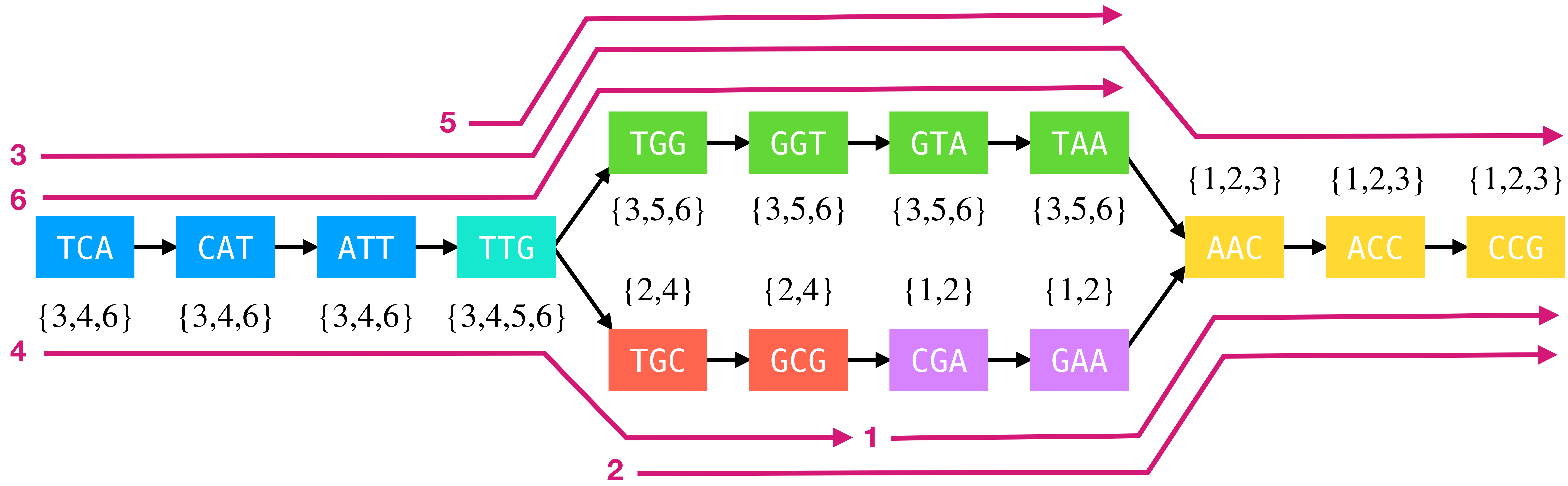
Colored *de Bruijn* graphs

- One-to-one correspondence between a set of k-mers and a *de Bruijn* graph (dBG).
- Example for k = 3 and **N = 6** references. References in $\mathscr{R}$ are spelled by **paths** in the graph.



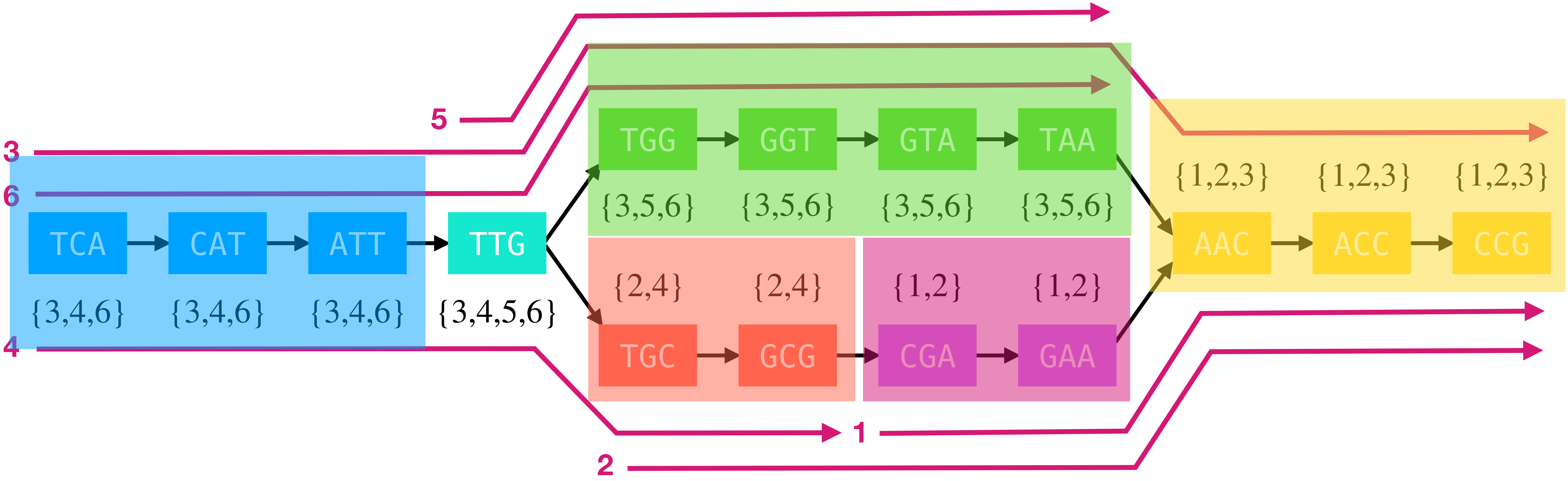
Colored *de Bruijn* graphs

- One-to-one correspondence between a set of k-mers and a *de Bruijn* graph (dBG).
- Example for k = 3 and **N = 6** references. References in $\mathscr{R}$ are spelled by **paths** in the graph.



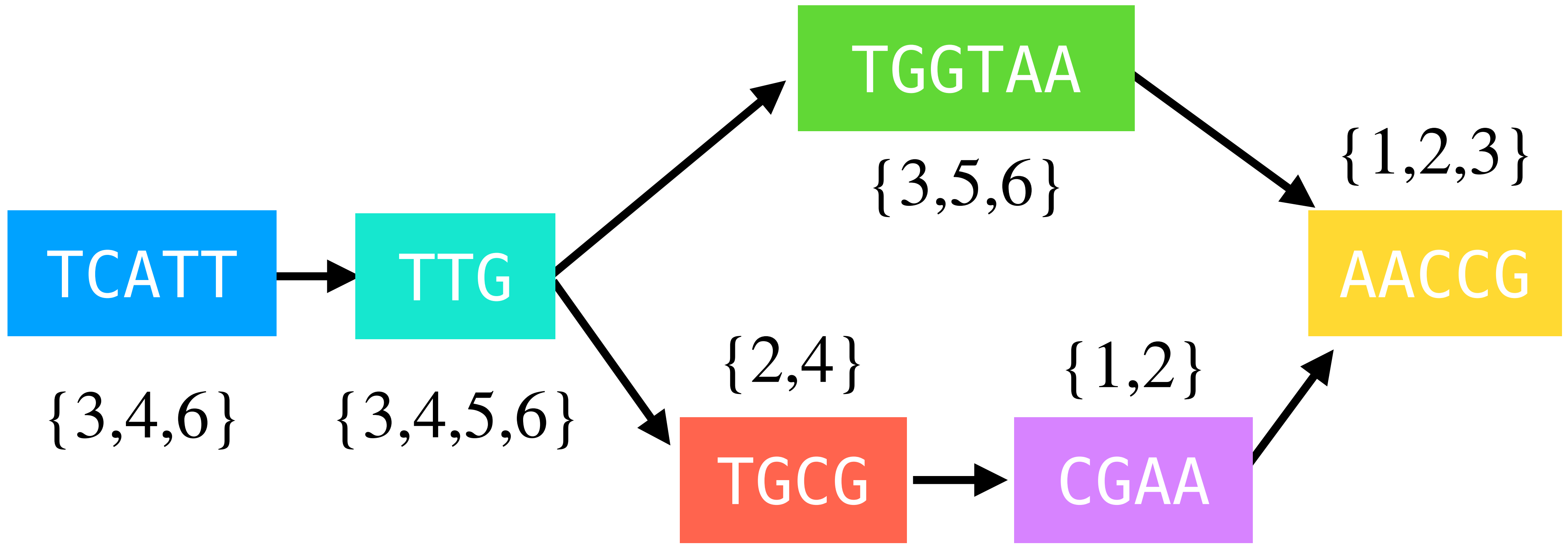
Colored *de Bruijn* graphs

- One-to-one correspondence between a set of k-mers and a *de Bruijn* graph (dBG).
- Example for k = 3 and **N = 6** references. References in $\mathscr{R}$ are spelled by **paths** in the graph.



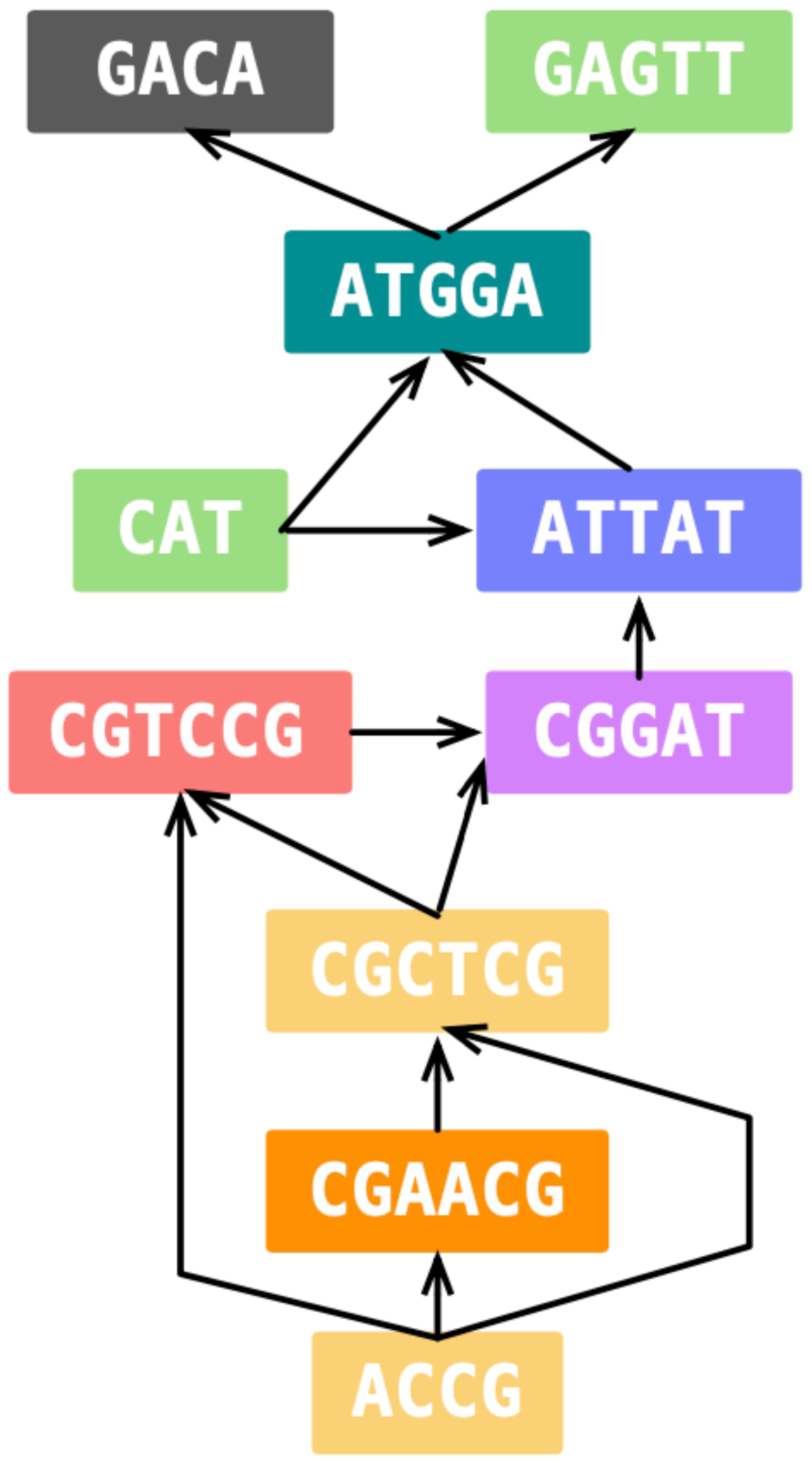
Colored compacted *de Bruijn* graphs

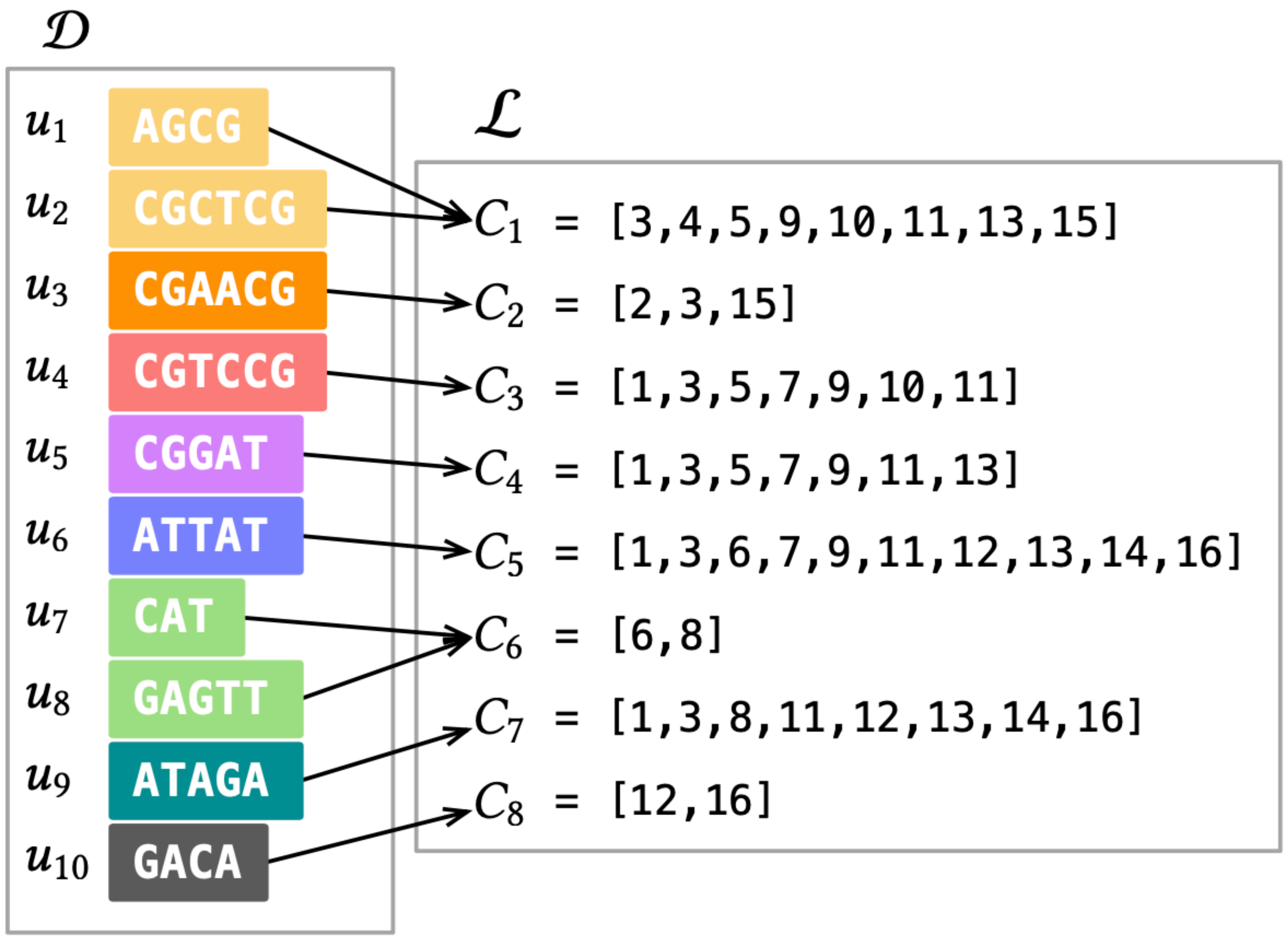
- One-to-one correspondence between a set of k-mers and a *de Bruijn* graph (dBG).
- Nodes having the **same color along non-branching paths** are collapsed into **unitigs**.



Colored compacted *de Bruijn* graphs

- Another, larger, example for **N = 16** references.





C_1 = [3,4,5,9,10,11,13,15]

C_2 = [2,3,15]

C_3 = [1,3,5,7,9,10,11]

C_4 = [1,3,5,7,9,11,13]

C_5 = [1,3,6,7,9,11,12,13,14,16]

C_6 = [6,8]

C_7 = [1,3,8,11,12,13,14,16]

C_8 = [12,16]

Properties of colored compacted dBGs

1. **Unitigs spell references in $\mathscr{R}$.** → We can represent the set of unitigs instead of the set of k-mers. Better space and cache locality.

2. **Unitigs are monochromatic.** → We store a color set for each unitig, rather than for each k-mer. We need an efficient map from k-mers to unitigs.

3. **Unitigs co-occur.** → Distinct unitigs often have the same color, i.e., they co-occur in the same subset of references. We have way less distinct colors than unitigs.
   We need an efficient map from unitigs to colors.

Properties of colored compacted dBGs

SSHash [P., 2022]

1. **Unitigs spell references in $\mathscr{R}$.** → We can represent the set of unitigs instead of the set of k-mers. Better space and cache locality.

2. **Unitigs are monochromatic.** → We store a color set for each unitig, rather than for each k-mer. We need an efficient map from k-mers to unitigs.

3. **Unitigs co-occur.** → Distinct unitigs often have the same color, i.e., they co-occur in the same subset of references. We have way less distinct colors than unitigs.
   We need an efficient map from unitigs to colors.

Properties of colored compacted dBGs

SSHash [P., 2022]

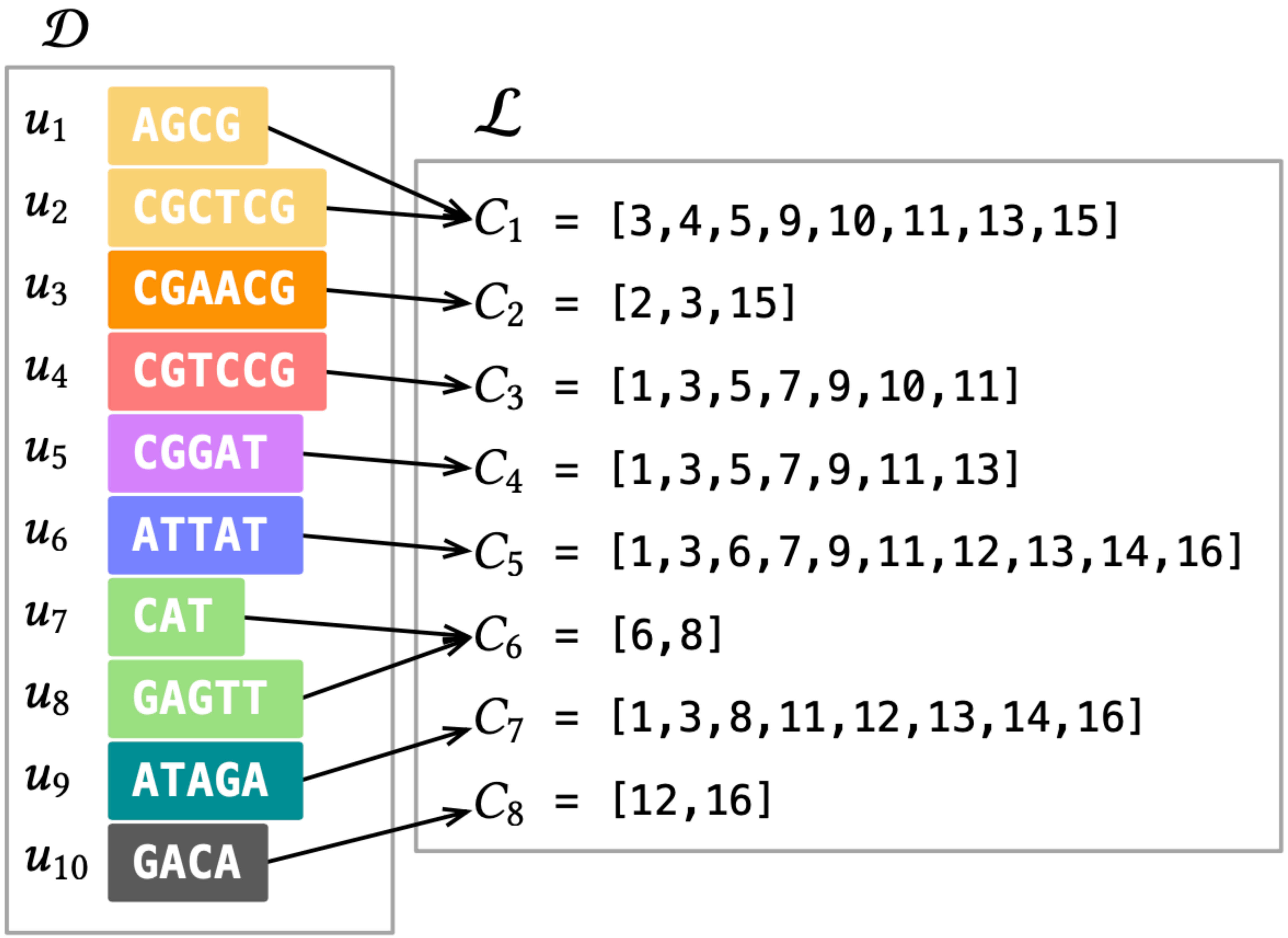
1. **Unitigs spell references in $\mathscr{R}$.** → We can represent the set of unitigs instead of the set of k-mers. Better space and cache locality.

2. **Unitigs are monochromatic.** → We store a color set for each unitig, rather than for each k-mer. We need an efficient map from k-mers to unitigs.

3. **Unitigs co-occur.** → Distinct unitigs often have the same color, i.e., they co-occur in the same subset of references. We have way less distinct colors than unitigs.
   We need an efficient map from unitigs to colors.

4. **Colors are similar when indexing pangenomes.** → Opportunity to achieve much better compression if colors are not compressed *individually* (each set independently of the others) but *common patterns are factored out and compressed once*.

Colors are similar when indexing pangenomes



$\mathcal{D}$

u_1 AGCG

u_2 CGCTCG

u_3 CGAACG

u_4 CGTCCG

u_5 CGGAT

u_6 ATTAT

u_7 CAT

u_8 GAGTT

u_9 ATAGA

u_{10} GACA

$\mathcal{L}$

C_1 = [3,4,5,9,10,11,13,15]

C_2 = [2,3,15]

C_3 = [1,3,5,7,9,10,11]

C_4 = [1,3,5,7,9,11,13]

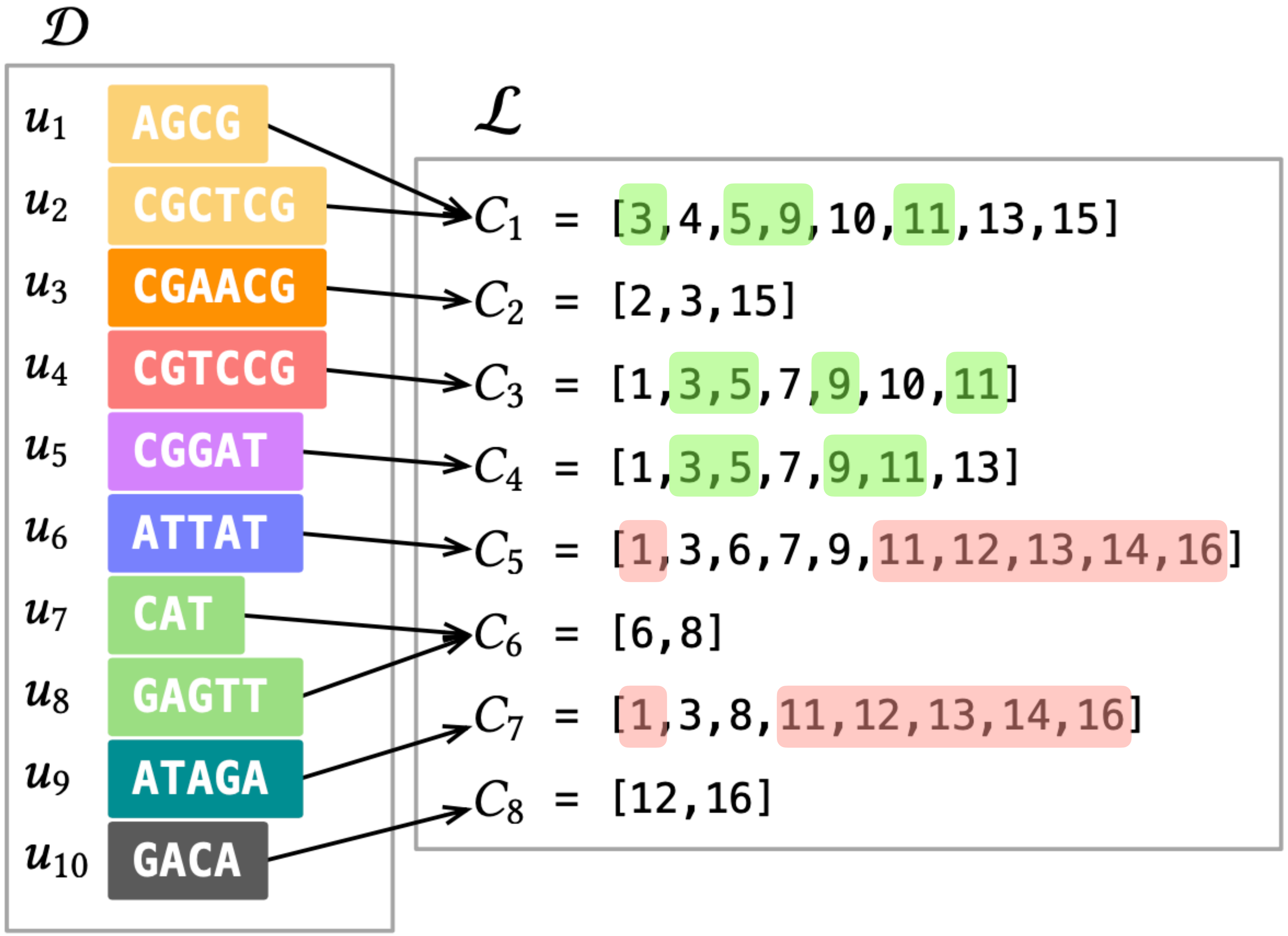
C_5 = [1,3,6,7,9,11,12,13,14,16]

C_6 = [6,8]

C_7 = [1,3,8,11,12,13,14,16]

C_8 = [12,16]

Colors are similar when indexing pangenomes



$\mathcal{D}$

u_1 AGCG

u_2 CGCTCG

u_3 CGAACG

u_4 CGTCCG

u_5 CGGAT

u_6 ATTAT

u_7 CAT

u_8 GAGTT

u_9 ATAGA

u_{10} GACA

$\mathcal{L}$

C_1 = [3,4,5,9,10,11,13,15]

C_2 = [2,3,15]

C_3 = [1,3,5,7,9,10,11]

C_4 = [1,3,5,7,9,11,13]

C_5 = [1,3,6,7,9,11,12,13,14,16]

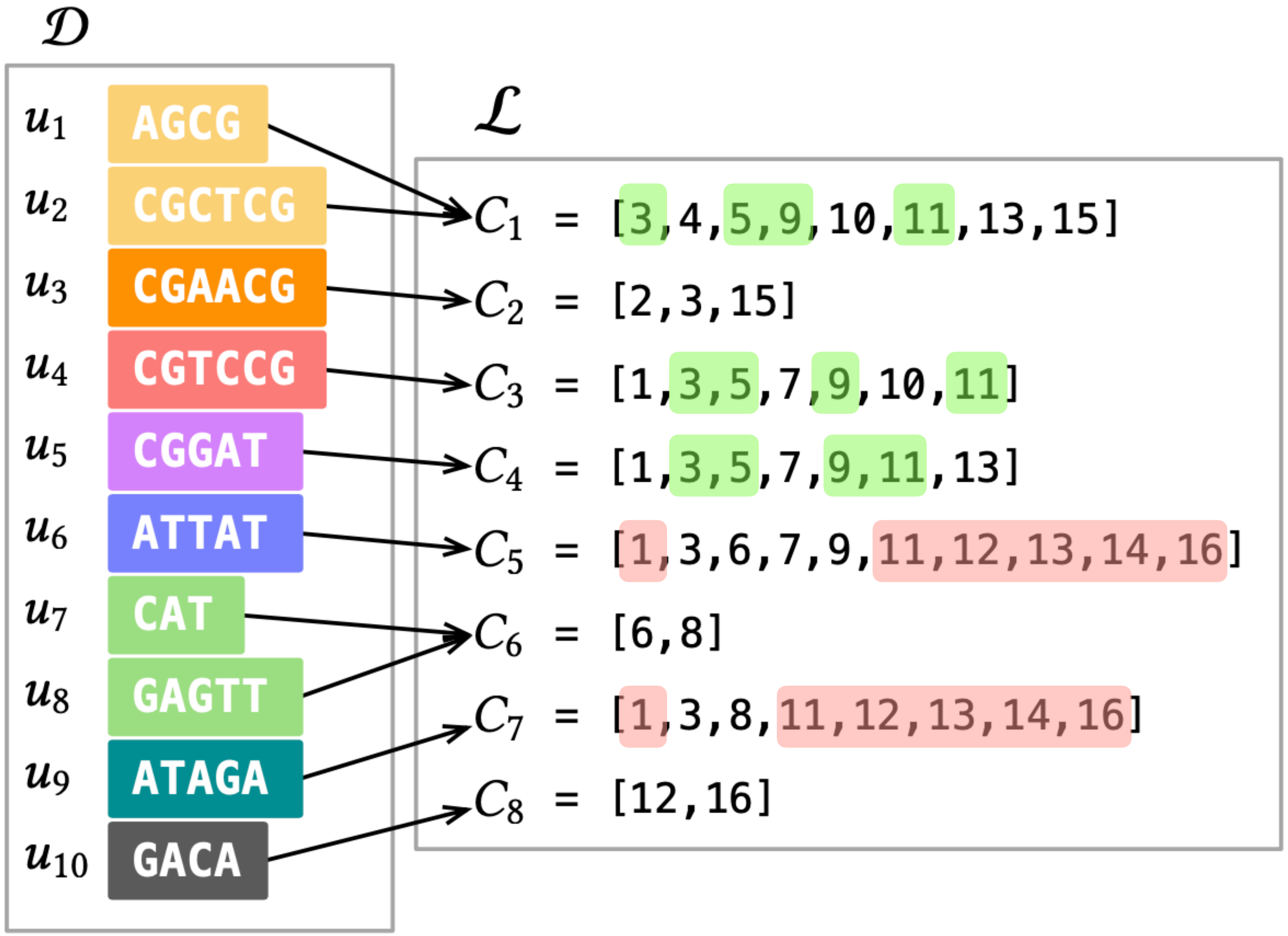
C_6 = [6,8]

C_7 = [1,3,8,11,12,13,14,16]

C_8 = [12,16]

- The pattern $\{3,5,9,11\}$ is currently represented three times.
- The pattern $\{1,11,12,13,14,16\}$ is represented twice.

Colors are similar when indexing pangenomes



C_1 = [3,4,5,9,10,11,13,15]

C_2 = [2,3,15]

C_3 = [1,3,5,7,9,10,11]

C_4 = [1,3,5,7,9,11,13]

C_5 = [1,3,6,7,9,11,12,13,14,16]

C_6 = [6,8]

C_7 = [1,3,8,11,12,13,14,16]

C_8 = [12,16]

- The pattern $\{3,5,9,11\}$ is currently represented three times.
- The pattern $\{1,11,12,13,14,16\}$ is represented twice.
- **Q.** How to factor out this redundancy?

Introducing *meta* and *partial* colors

- Determine a **partition** of $[N] = \{1,\ldots,N\}$ so that references in the same partition are *similar*. Take the set of distinct **partial** colors for each partition → factor out the redundancy.

- **Intuition:** similar references induce similar colors and thus *share patterns*.

- Now we can render each original color as a sequence of references, or **meta** colors, to those **partial** colors.

- **Bonus:** Reference identifiers inside each partition can be re-assigned to save space.

- Let's consider an example.

Introducing *meta* and *partial* colors

- Example for **N = 16** references and **4** partitions.

$$\begin{array}{r} \{\ 1\ 12\ 13\ 14\ 16\ \}\{\ 3\ 5\ 9\ \}\{\ 7\ 11\ \}\{\ 2\ \ 4\ \ 6\ \ 8\ \ 10\ 15\ \} \\ \textit{new identifiers} \rightarrow 1\ \ 2\ \ 3\ \ 4\ \ 5\ \quad\ 6\ 7\ 8\ \quad\ 9\ 10\ \quad\ 11\ 12\ 13\ 14\ 15\ 16\ \ \end{array} \longleftarrow$$

this defines a permutation π

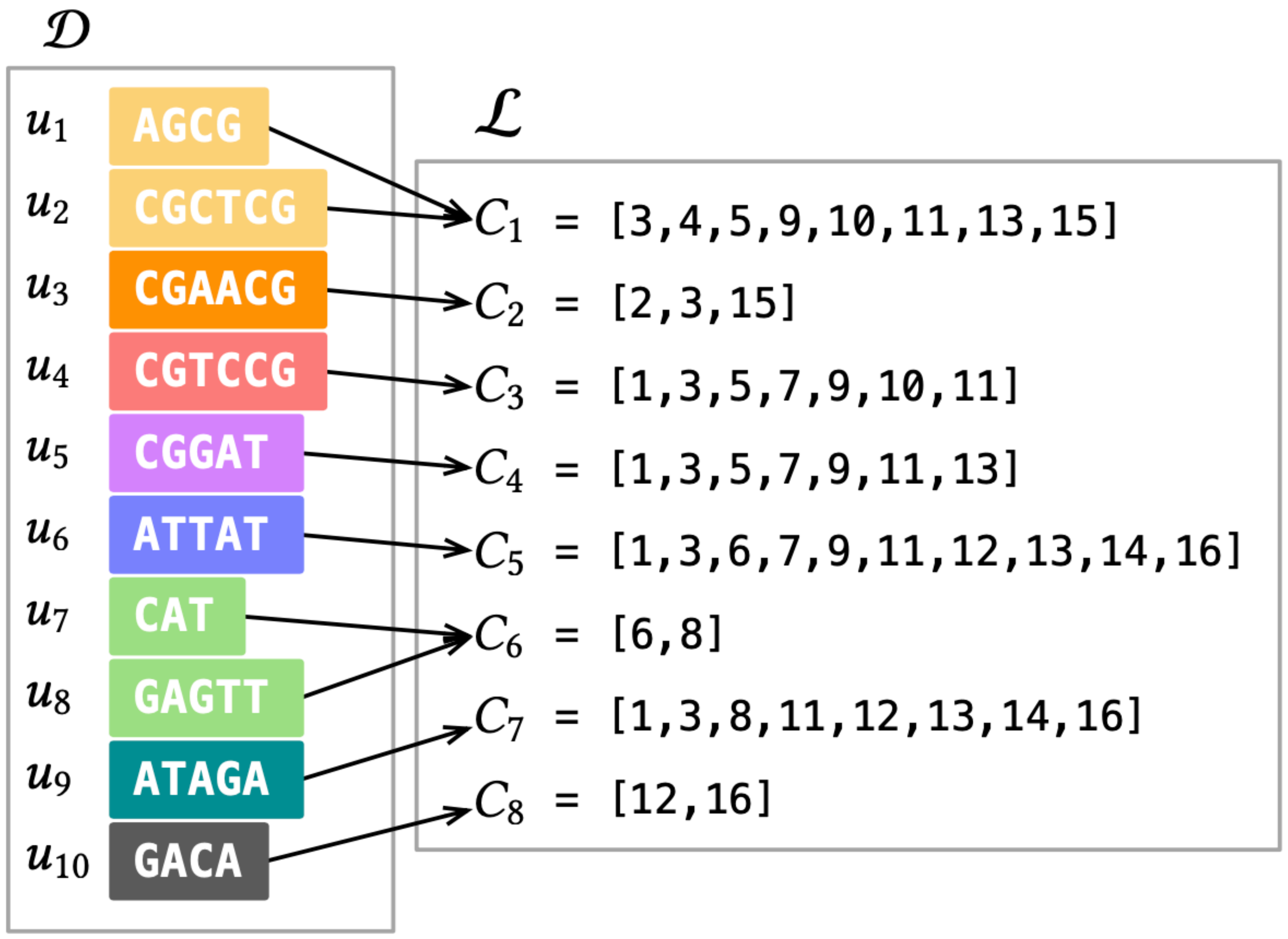
Introducing *meta* and *partial* colors

- Example for **N = 16** references and **4** partitions.

$$\{\ 1\ 12\ 13\ 14\ 16\ \}\{\ 3\ 5\ 9\ \}\{\ 7\ 11\ \}\{\ 2\ \ 4\ \ 6\ \ 8\ \ 10\ 15\ \}$$

$$\textit{new identifiers} \rightarrow 1\ \ 2\ \ 3\ \ 4\ \ 5\qquad 6\ 7\ 8\qquad 9\ 10\qquad 11\ 12\ 13\ 14\ 15\ 16$$

this defines a permutation π



$\mathcal{D}$

u_1 AGCG

u_2 CGCTCG

u_3 CGAACG

u_4 CGTCCG

u_5 CGGAT

u_6 ATTAT

u_7 CAT

u_8 GAGTT

u_9 ATAGA

u_{10} GACA

$\mathcal{L}$

C_1 = [3,4,5,9,10,11,13,15]

C_2 = [2,3,15]

C_3 = [1,3,5,7,9,10,11]

C_4 = [1,3,5,7,9,11,13]

C_5 = [1,3,6,7,9,11,12,13,14,16]

C_6 = [6,8]

C_7 = [1,3,8,11,12,13,14,16]

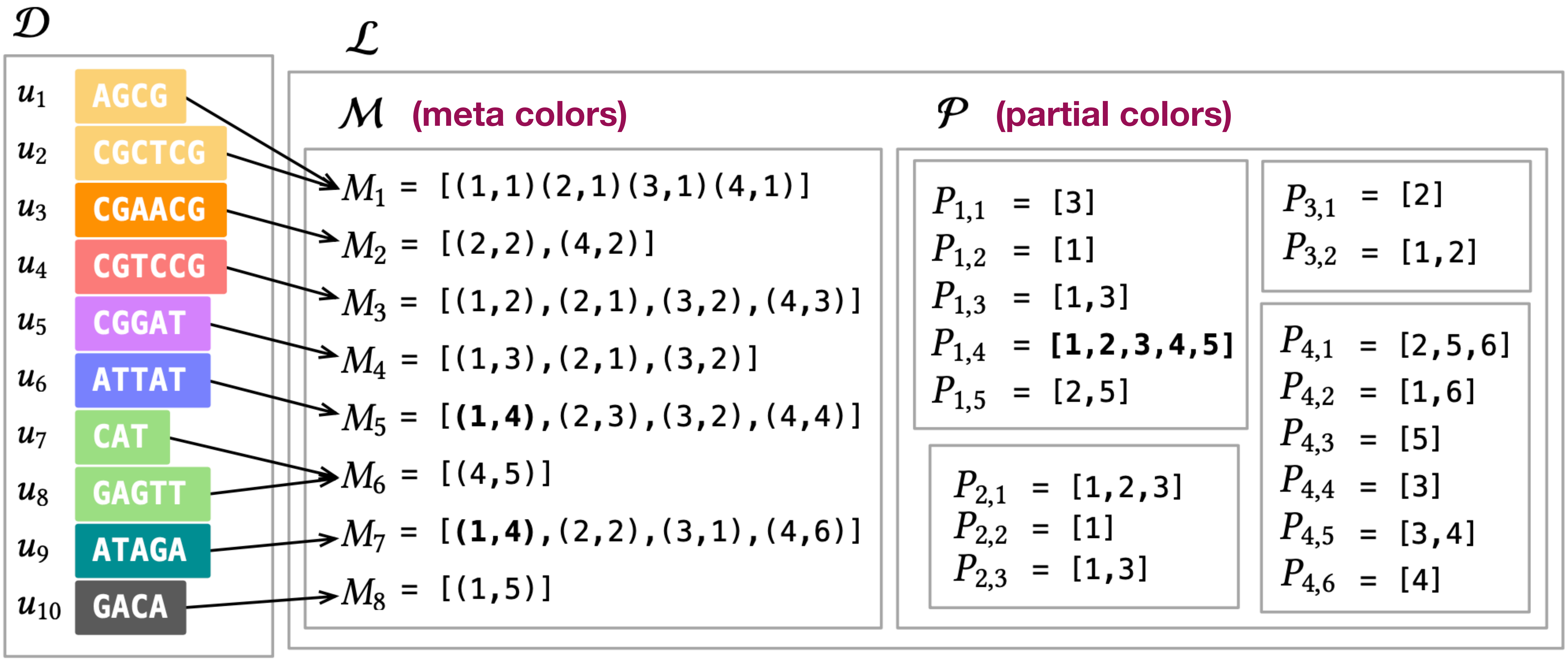
C_8 = [12,16]

$\xrightarrow{\pi}$

$$C_1 = [3|6,7,8|10|12,15,16]$$
$$C_2 = [6|11,16]$$
$$C_3 = [1|6,7,8|9,10|15]$$
$$C_4 = [1,3|6,7,8|9,10]$$
$$C_5 = [1,2,3,4,5|6,8|9,10|13]$$
$$C_6 = [13,14]$$
$$C_7 = [1,2,3,4,5|6|10|14]$$
$$C_8 = [2,5]$$

Introducing *meta* and *partial* colors

- Example for **N = 16** references and **4** partitions.



$\mathcal{D}$

u_1 AGCG

u_2 CGCTCG

u_3 CGAACG

u_4 CGTCCG

u_5 CGGAT

u_6 ATTAT

u_7 CAT

u_8 GAGTT

u_9 ATAGA

u_{10} GACA

$\mathcal{L}$

$\mathcal{M}$ (meta colors)

M_1 = [(1,1)(2,1)(3,1)(4,1)]

M_2 = [(2,2),(4,2)]

M_3 = [(1,2),(2,1),(3,2),(4,3)]

M_4 = [(1,3),(2,1),(3,2)]

M_5 = [**(1,4)**,(2,3),(3,2),(4,4)]

M_6 = [(4,5)]

M_7 = [**(1,4)**,(2,2),(3,1),(4,6)]

M_8 = [(1,5)]

$\mathcal{P}$ (partial colors)

$P_{1,1}$ = [3]

$P_{1,2}$ = [1]

$P_{1,3}$ = [1,3]

$P_{1,4}$ = **[1,2,3,4,5]**

$P_{1,5}$ = [2,5]

$P_{2,1}$ = [1,2,3]

$P_{2,2}$ = [1]

$P_{2,3}$ = [1,3]

$P_{3,1}$ = [2]

$P_{3,2}$ = [1,2]

$P_{4,1}$ = [2,5,6]

$P_{4,2}$ = [1,6]

$P_{4,3}$ = [5]

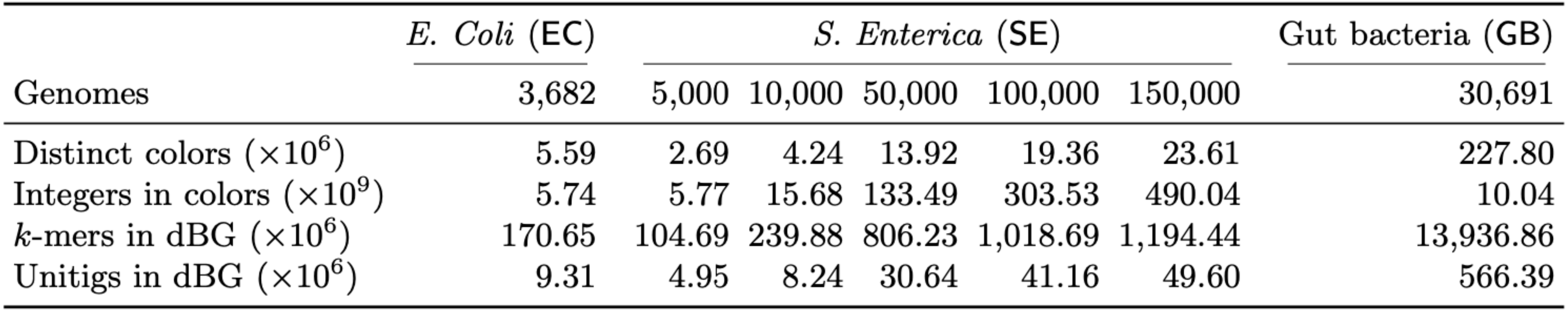
$P_{4,4}$ = [3]

$P_{4,5}$ = [3,4]

$P_{4,6}$ = [4]

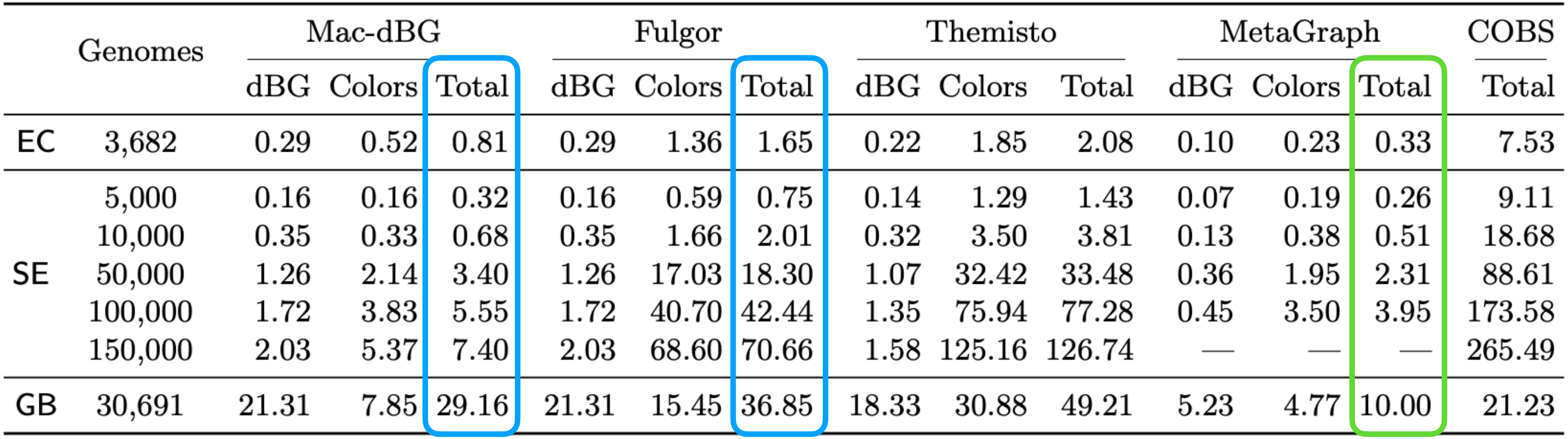
Results

- Results on some large pangenome collections of different complexities.



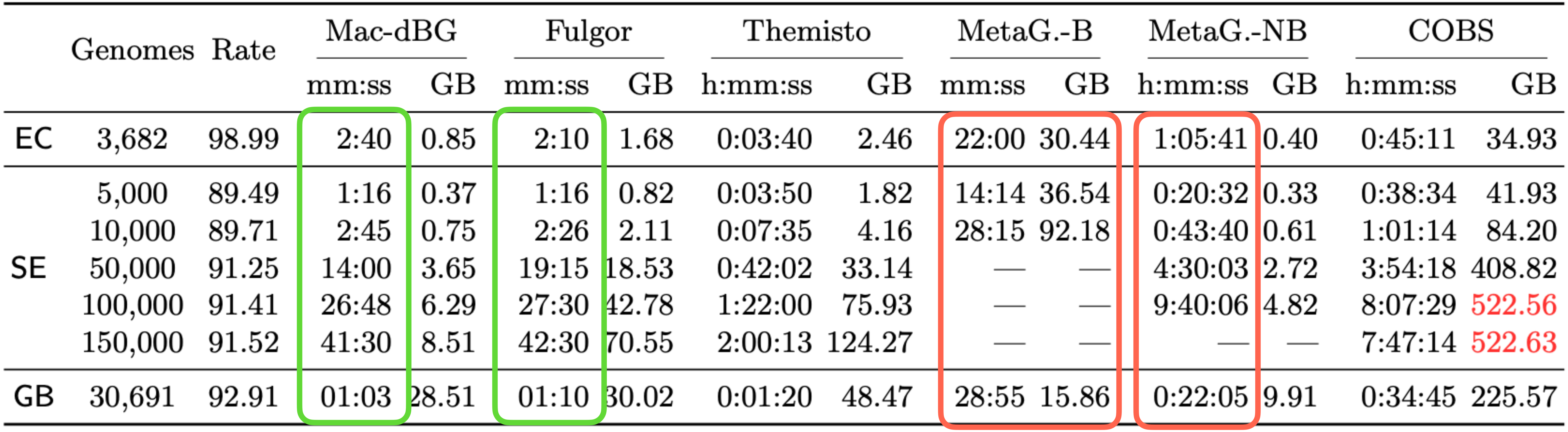
	E. Coli (EC)	*S. Enterica* (SE)					Gut bacteria (GB)
Genomes	3,682	5,000	10,000	50,000	100,000	150,000	30,691
Distinct colors ($\times 10^6$)	5.59	2.69	4.24	13.92	19.36	23.61	227.80
Integers in colors ($\times 10^9$)	5.74	5.77	15.68	133.49	303.53	490.04	10.04
k-mers in dBG ($\times 10^6$)	170.65	104.69	239.88	806.23	1,018.69	1,194.44	13,936.86
Unitigs in dBG ($\times 10^6$)	9.31	4.95	8.24	30.64	41.16	49.60	566.39

Index space in GB



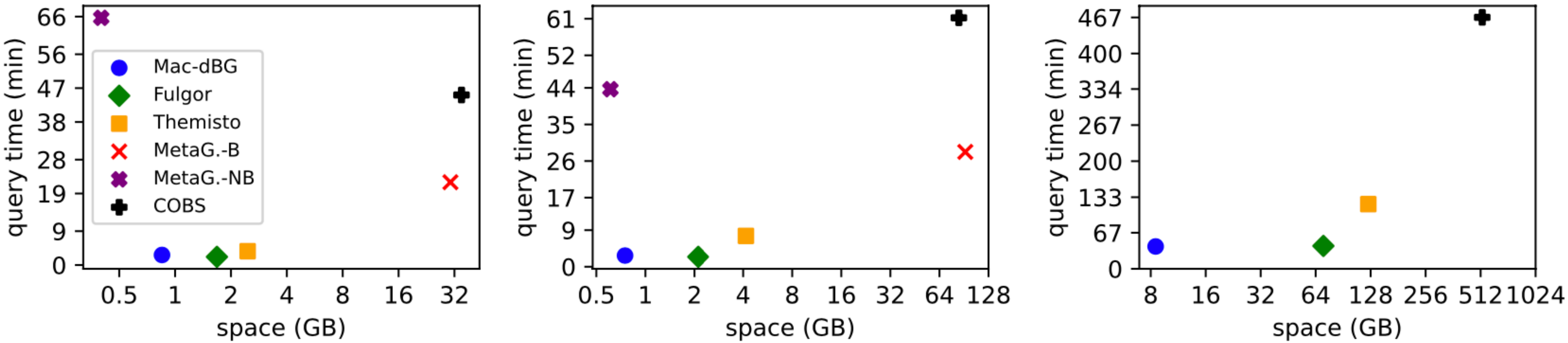
	Genomes	Mac-dBG			Fulgor			Themisto			MetaGraph			COBS
		dBG	Colors	Total	dBG	Colors	Total	dBG	Colors	Total	dBG	Colors	Total	Total
EC	3,682	0.29	0.52	0.81	0.29	1.36	1.65	0.22	1.85	2.08	0.10	0.23	0.33	7.53
SE	5,000	0.16	0.16	0.32	0.16	0.59	0.75	0.14	1.29	1.43	0.07	0.19	0.26	9.11
	10,000	0.35	0.33	0.68	0.35	1.66	2.01	0.32	3.50	3.81	0.13	0.38	0.51	18.68
	50,000	1.26	2.14	3.40	1.26	17.03	18.30	1.07	32.42	33.48	0.36	1.95	2.31	88.61
	100,000	1.72	3.83	5.55	1.72	40.70	42.44	1.35	75.94	77.28	0.45	3.50	3.95	173.58
	150,000	2.03	5.37	7.40	2.03	68.60	70.66	1.58	125.16	126.74	—	—	—	265.49
GB	30,691	21.31	7.85	29.16	21.31	15.45	36.85	18.33	30.88	49.21	5.23	4.77	10.00	21.23

Query efficiency



	Genomes	Rate	Mac-dBG		Fulgor		Themisto		MetaG.-B		MetaG.-NB		COBS	
			mm:ss	GB	mm:ss	GB	h:mm:ss	GB	mm:ss	GB	h:mm:ss	GB	h:mm:ss	GB
EC	3,682	98.99	2:40	0.85	2:10	1.68	0:03:40	2.46	22:00	30.44	1:05:41	0.40	0:45:11	34.93
SE	5,000	89.49	1:16	0.37	1:16	0.82	0:03:50	1.82	14:14	36.54	0:20:32	0.33	0:38:34	41.93
	10,000	89.71	2:45	0.75	2:26	2.11	0:07:35	4.16	28:15	92.18	0:43:40	0.61	1:01:14	84.20
	50,000	91.25	14:00	3.65	19:15	18.53	0:42:02	33.14	—	—	4:30:03	2.72	3:54:18	408.82
	100,000	91.41	26:48	6.29	27:30	42.78	1:22:00	75.93	—	—	9:40:06	4.82	8:07:29	522.56
	150,000	91.52	41:30	8.51	42:30	70.55	2:00:13	124.27	—	—	—	—	7:47:14	522.63
GB	30,691	92.91	01:03	28.51	01:10	30.02	0:01:20	48.47	28:55	15.86	0:22:05	9.91	0:34:45	225.57

Overall comparison



(a) EC

(b) SE 10,000

(c) SE 150,000

Conclusions

- The meta-colored de Bruijn graph (or Mac-dBG) embodies a **superior space/time trade-off** compared to the state of the art.
- **Space improvement can be dramatic**: almost up to one order of magnitude.
- **Query efficiency not harmed** despite of the significant space reduction.
- **Take-away:** No reason not to use meta-colored dBGs to compress and index pangenomes!
- **Code** is available: https://github.com/jermp/fulgor/releases/tag/v2.0.0.
- Many **open problems and future directions**, check our pre-print out: https://doi.org/10.1101/2023.07.21.550101.

Conclusions

- The meta-colored de Bruijn graph (or Mac-dBG) embodies a **superior space/time trade-off** compared to the state of the art.

- **Space improvement can be dramatic**: almost up to one order of magnitude.

- **Query efficiency not harmed** despite of the significant space reduction.

- **Take-away:** No reason not to use meta-colored dBGs to compress and index pangenomes!

- **Code** is available: https://github.com/jermp/fulgor/releases/tag/v2.0.0.

- Many **open problems and future directions**, check our pre-print out: https://doi.org/10.1101/2023.07.21.550101.

Thank you for the attention!