Assembly-free detection of CRISPR-Cas systems in metagenomes through HMM-guided search in de Bruijn graphs

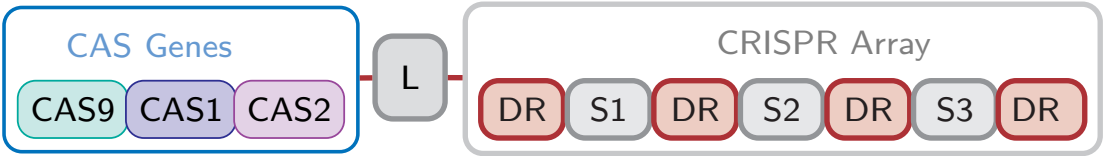
Progress Report

RNA Biology & Bioinformatics, Uni Stuttgart

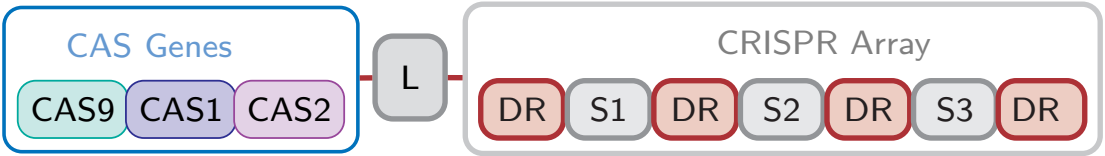
February 2026

Outline

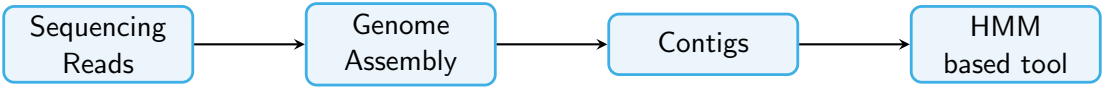
What is CRISPR-Cas?



What is CRISPR-Cas?



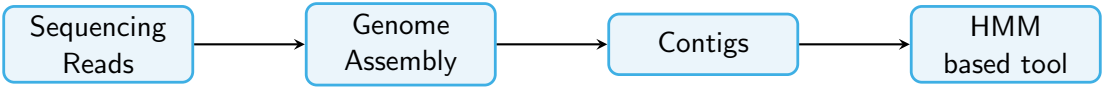
Metagenomic Datasets



Goal: Requires complete genome assembly

- Short sequencing reads - 125 bps, genes - from 150bps ⇒ not feasible
- Contigs - short, upto 1000s bps ⇒ not enough for casettes

Metagenomic Datasets

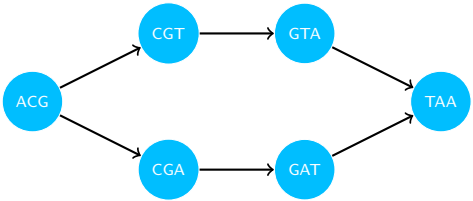


Goal: Requires complete genome assembly

- Short sequencing reads - 125 bps, genes - from 150bps ⇒ not feasible
- Contigs - short, upto 1000s bps ⇒ not enough for casettes

Our approach: Skip assembly, work directly on the **de Bruijn graph**

What is a de Bruijn Graph?



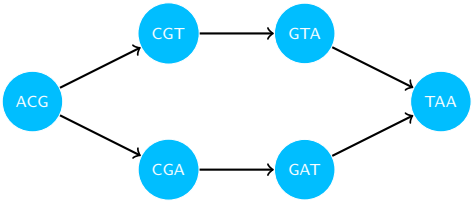
Nodes: N k-mers (DNA words)

Edges: E - overlaps of $k-1$

Paths: p_i - possible sequences

Real world: k=23, $|E| = [1-18]$ billion, multiple combinations of paths

What is a de Bruijn Graph?



Nodes: N k-mers (DNA words)

Edges: E - overlaps of $k-1$

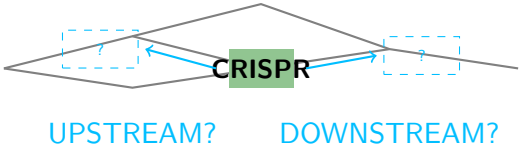
Paths: p_i - possible sequences

Real world: k=23, $|E| = [1-18]$ billion, multiple combinations of paths

Challenge: Branchings and scale create multiple noisy paths

- Heuristics that do not compromise the accuracy
- Algorithm that corresponds to modern standards
- Elegant to define and fun to develop

Our Goal



Given:

- A de Bruijn graph from metagenomic sequencing
- Location of a CRISPR repeat (found by *MCAAT*)

Find:

- All CAS genes in both directions (upstream and downstream)
- Classify the cassette type (I-A, I-E, II-A, III-B, etc.)

How Do We Recognize a CAS Gene?

Hidden Markov Models (HMMs) — statistical profiles of protein families

HMM Profile

Position-specific scores

align ⟶

Query Sequence

`MKTLLVGNTGSGKS...`

Score: 450 bits

⇒ **Cas1 gene**

The process:

1. Find a potential START codon (ATG, GTG, TTG) in the graph
2. Translate DNA → protein (following the graph)
3. Score against HMM profile
4. High score = CAS gene match

The Search Space Problem

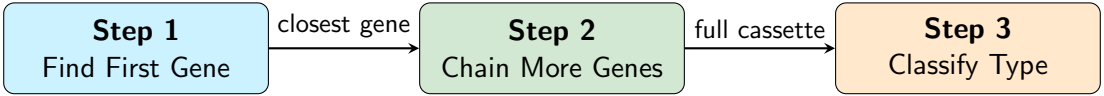
Naive approach:

- For each START codon candidate
- Test against each of ∼22 first-gene HMM profiles
- Each HMM scan: ∼12 ms (translate + align)

Worst case: $15,000 \times 22 \times 12\text{ms} =$ **66 minutes per cassette**

This is unacceptable for large-scale metagenomics!

Algorithm Overview



1. **First Gene Detection:** Find the CAS gene closest to the CRISPR repeat
   - Search both upstream and downstream
   - This is the computational bottleneck
2. **Gene Chaining:** Starting from first gene, find subsequent genes
   - Exploit operon structure (genes are adjacent)
3. **Type Classification:** Match gene combination to known CAS types

Step 1: First Gene Detection

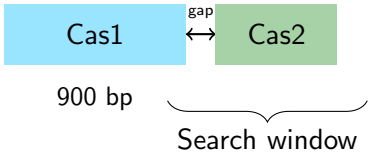
Algorithm 1 Find First CAS Gene

Find all START codons within distance $[d_{\text{min}}, d_{\text{max}}]$ from repeat
Sort candidates by distance (closest first)
for each candidate c **do**
    for each first-gene HMM profile p **do**
        Translate c following graph, score against p
        if normalized_score $\geq \tau_{\text{norm}}$ **then**
            return gene match
        end if
    end for
end for
return no gene found

Key parameter: $\tau_{\text{norm}} = 0.30$ bits per HMM position

Step 2: Gene Chaining (PLTS)

Profile-Length-Targeted Search — exploit operon biology



Key insight: Operon genes are tightly packed!

- Intergenic gap typically -24 to $+300$ bp
- If Gene 1 is 900bp, Gene 2 starts around position 900 $\pm$ small gap
- Don't search everywhere — search where genes *should* be

Result: Candidates reduced from thousands to $\sim$50–100 per gene

Step 3: Type Classification

Type I-E requires:
✓ cas1, cas2
✓ cas3 (helicase)
✓ cse1 (cascade)

Classification rules: From published CAS type definitions

- Each type has mandatory genes
- Some have optional/accessory genes
- Match found genes against rules ⇒ determine type

The Speed Problem

Observation: UPSTREAM and DOWNSTREAM behave differently

UPSTREAM

- CAS genes usually exist
- Found at candidate ∼100
- Early-stop works!
- ∼45 seconds

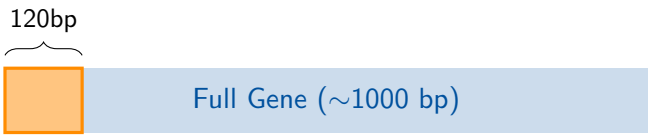
DOWNSTREAM

- CAS genes often *don't* exist
- Must scan all 15,000 candidates
- No early-stop possible
- ∼66 minutes

Problem: When there's no gene, we waste time proving it doesn't exist.

Solution: Pre-filter candidates with a *shallow* HMM scan

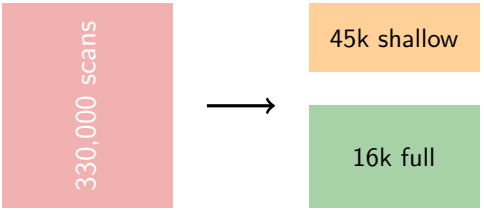
Optimization: Shallow Filter



Why it works:

- Random sequence: no HMM signal even in 40 amino acids
- Real CAS gene: detectable signal in N-terminal region
- Threshold 0.05 $\ll$ 0.30 (conservative, avoids false negatives)

Optimization Results



Scenario	Before	After
UPSTREAM (gene exists)	45 sec	45 sec
DOWNSTREAM (gene exists)	66 min	$<$1 min
DOWNSTREAM (no gene)	66 min	$\sim$5 min

Key Parameters

Distance Limits

d_{min}	50 bp (skip trailer)
d_{max} UP	5000 bp
d_{max} DOWN	1000 bp
g_{min}	-24 bp (overlap OK)
g_{max}	300 bp (max gap)

Scoring Thresholds

τ_{norm}	0.30 bits/pos
τ_{shallow}	0.05 bits/pos
τ_{chain}	0.10 bits/pos

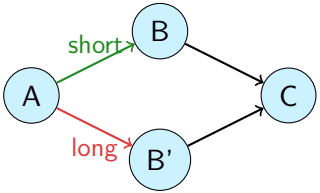
Note: Thresholds are heuristic and require empirical validation

First-gene profiles: Dynamically computed from type rules ($\sim$22 profiles)

- Includes: cas1, cas2, cas3, cas5, cas6, cas7, cas8, cas10, cas12, cas13...

Why HMM Signal, Not Length?

Rejected idea: Filter candidates by estimated ORF length



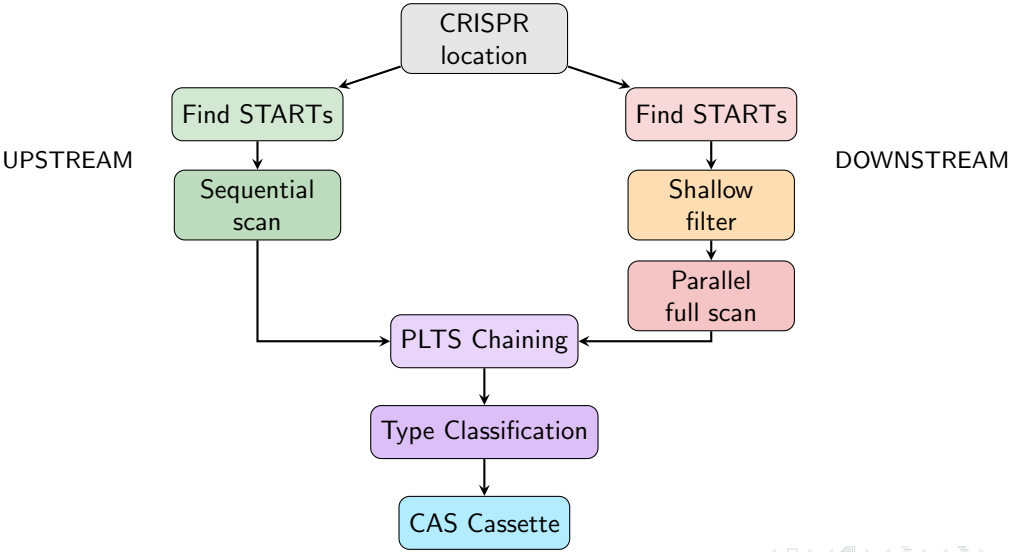
Problem:
ORF length follows ONE path.
True gene may be on
a different path!

$\Rightarrow$ False negatives

Correct approach: Filter by HMM signal

- The HMM aligns to the actual sequence, following the true path
- Shallow scan uses same HMM, just shorter
- **HMM is ground truth** — geometry estimates are unreliable

Complete Pipeline



Key Takeaways

1. **Graph-based detection**
   - Work directly on de Bruijn graph, skip assembly
   - Potential for fragmented metagenomes
2. **Two-phase search**
   - First gene: exhaustive search near repeat
   - Chaining: targeted search using operon geometry (PLTS)
3. **Direction-aware optimization**
   - UPSTREAM: early-stop (gene usually exists)
   - DOWNSTREAM: shallow filter (gene often absent)
4. **HMM is ground truth**
   - Filter by signal, not geometry
   - Handles graph ambiguity correctly

Limitations & Future Work

Current limitations:

- Scoring thresholds are heuristic (not calibrated via ROC)
- Shallow filter depth (120bp) not validated across CAS families
- No systematic benchmark against CRISPRCasFinder/CasTyper

Future work:

- Threshold calibration using CRISPRCasDB annotations
- Benchmark on fragmented metagenomes vs. contig-based tools
- Demonstrate cases where graph approach finds cassettes that assembly misses

Questions?