

3 - A simple genome assembly

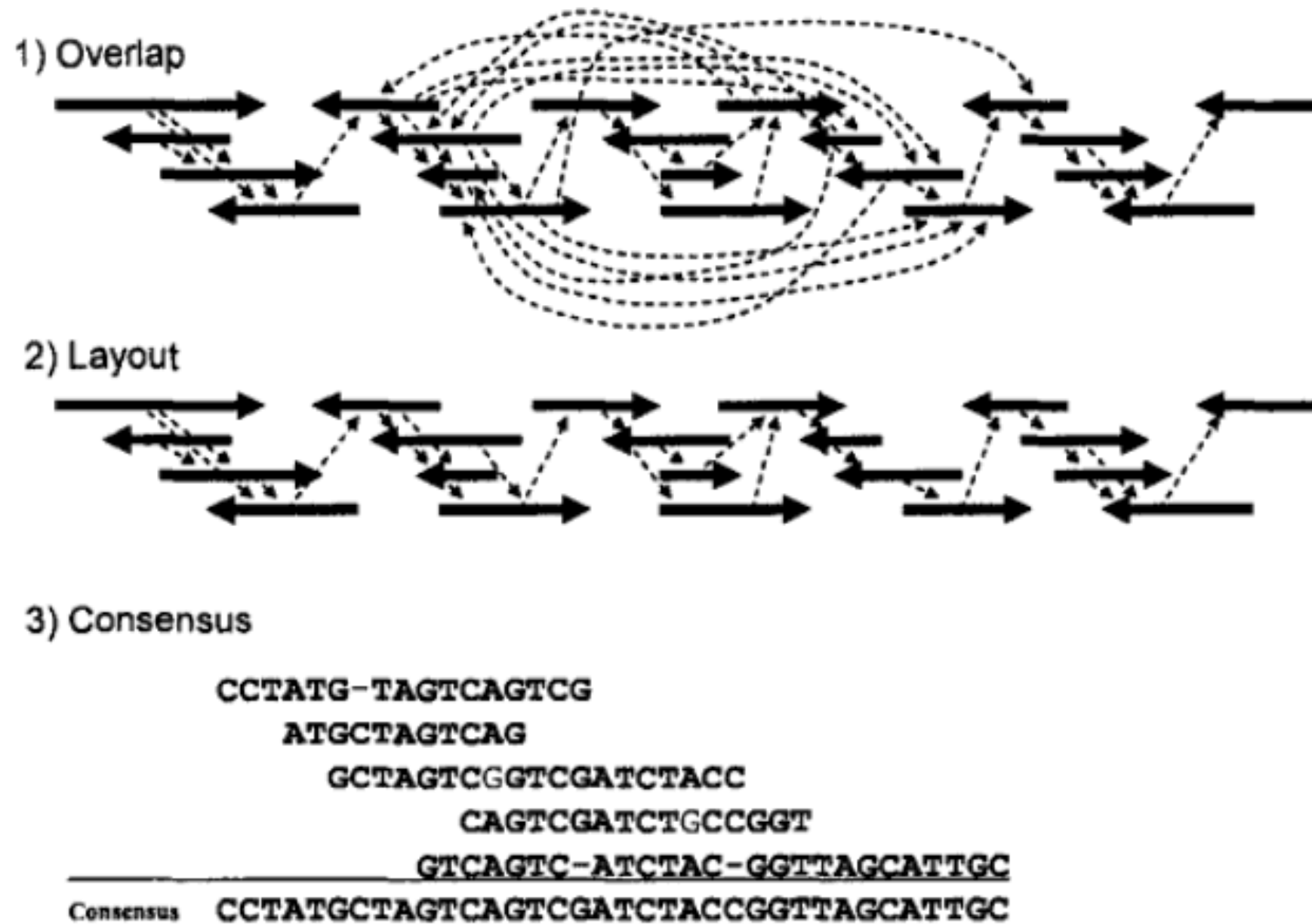
Wednesday morning

Bernardo J. Clavijo
Richard Smith-Unna
Gonzalo Garcia



Overlap Layout Consensus

Overlap - Layout - Consensus



Overlap Layout Consensus: Key points

- Finding overlaps and defining them is key.
- The layout can be quite difficult.
- The method tracks every read.
- The consensus is constructed from the reads.

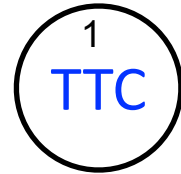
De Bruijn Graphs

Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

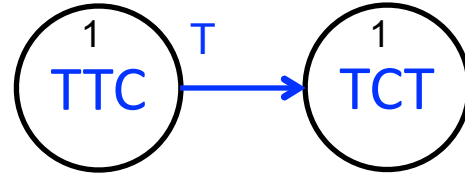
Assembling a De Bruijn Graph

```
>seq1  
TTC TAAGT  
>seq2  
CGATTCTA
```



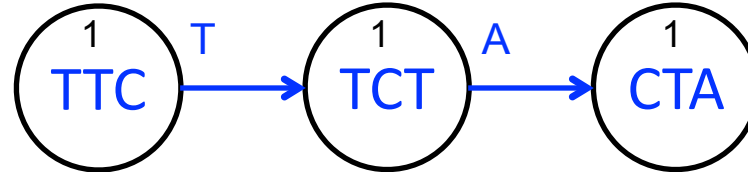
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



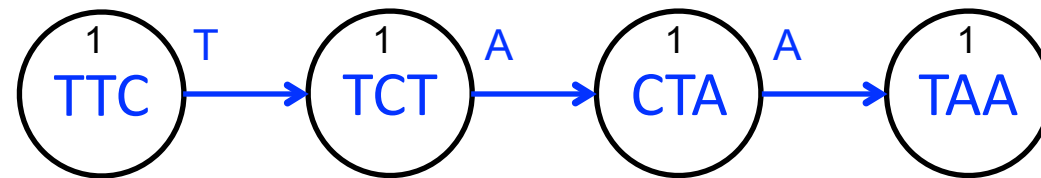
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



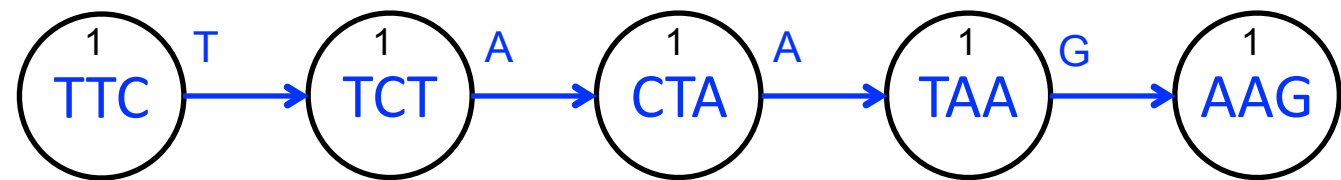
Assembling a De Bruijn Graph

```
>seq1  
TTC1TAAAGT  
>seq2  
CGATTCTA
```



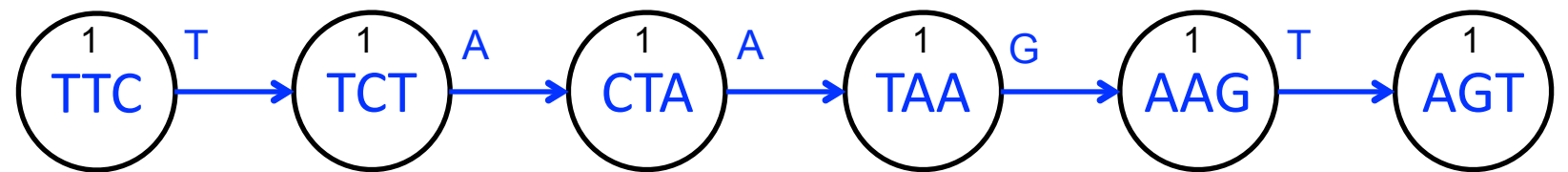
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



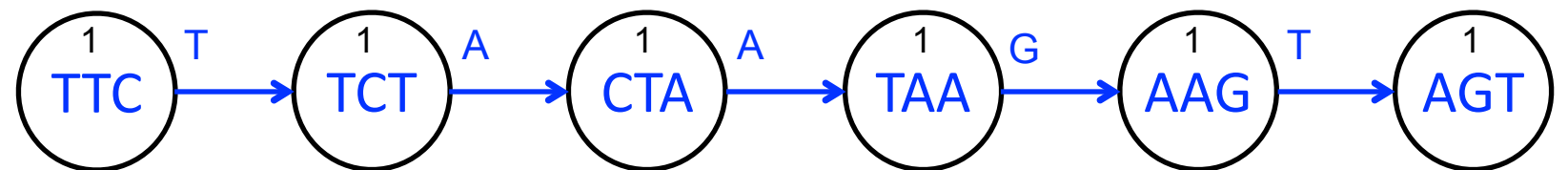
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



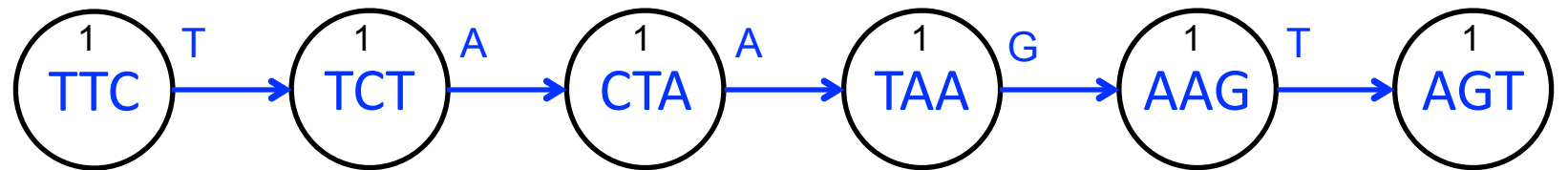
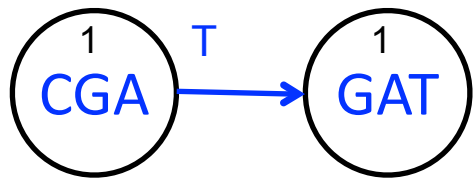
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



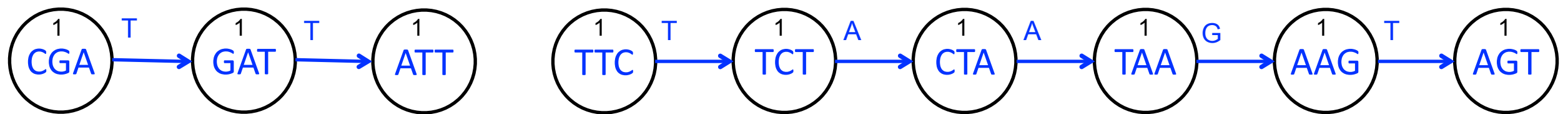
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCCTA
```



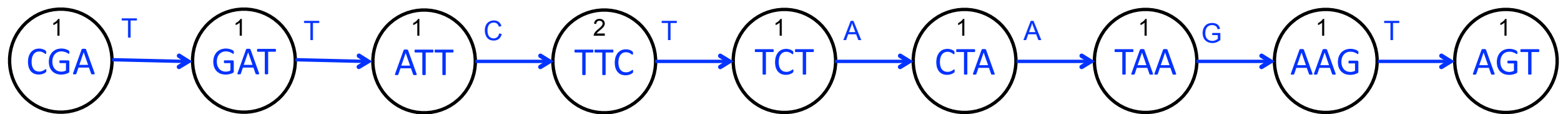
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



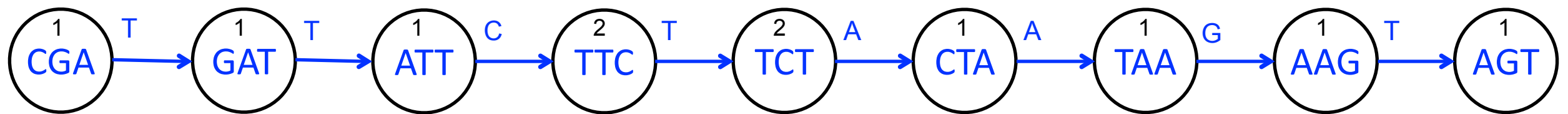
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



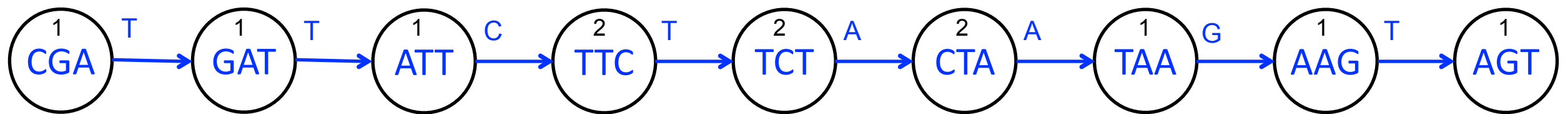
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



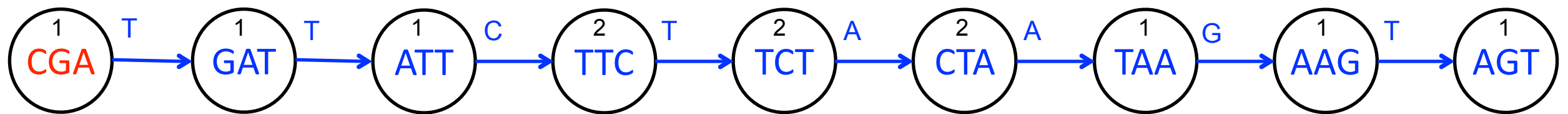
Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



Assembling a De Bruijn Graph

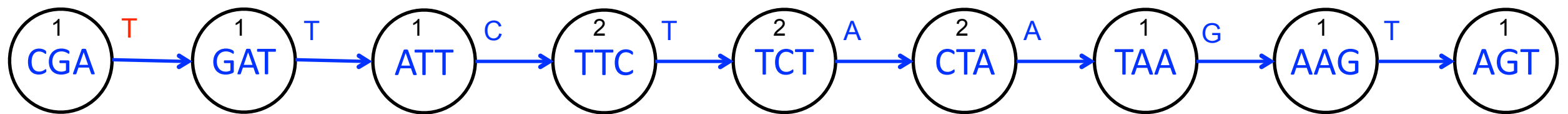
```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



CGA

Assembling a De Bruijn Graph

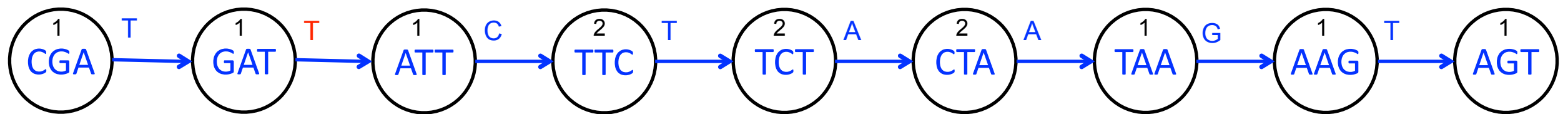
```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



CGAT

Assembling a De Bruijn Graph

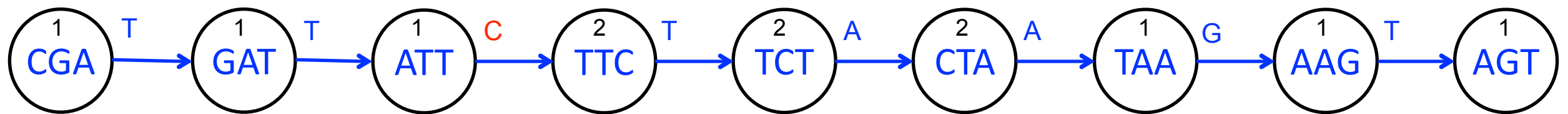
```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



CGATT

Assembling a De Bruijn Graph

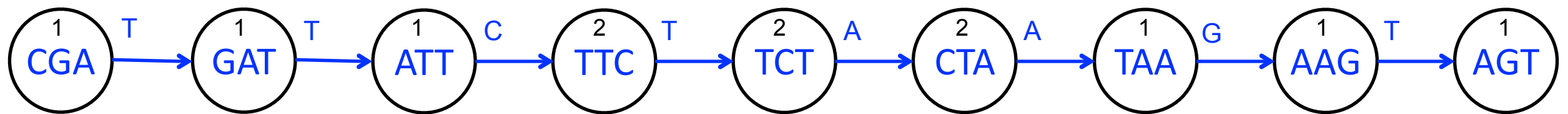
```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```



CGATT**C**

Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

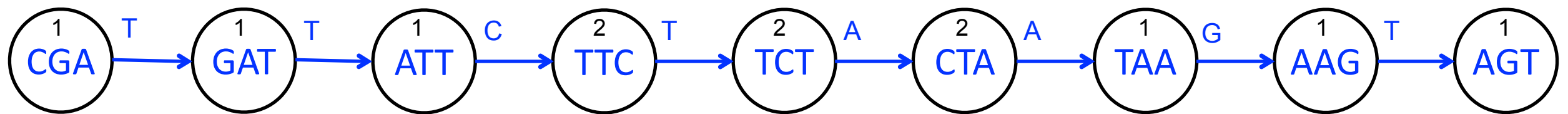


CGATTCTAAGT

Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

```
>seq3  
CGATTGTAAGT
```

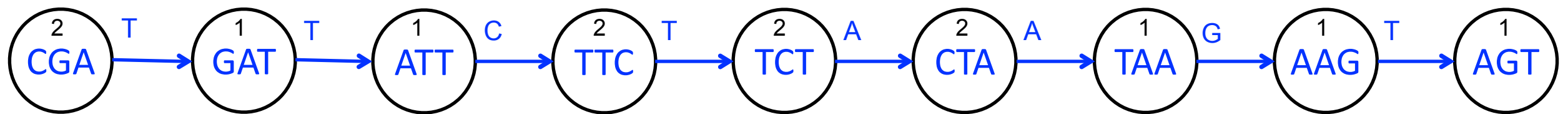


CGATTCTAAGT

Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

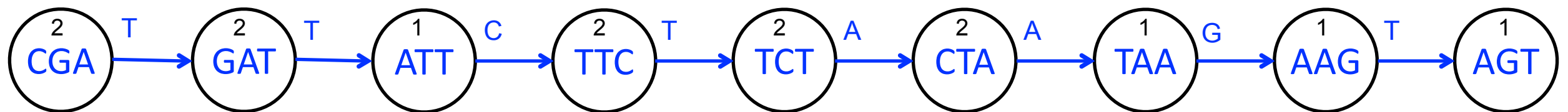
```
>seq3  
CGATTGTAAGT
```



Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

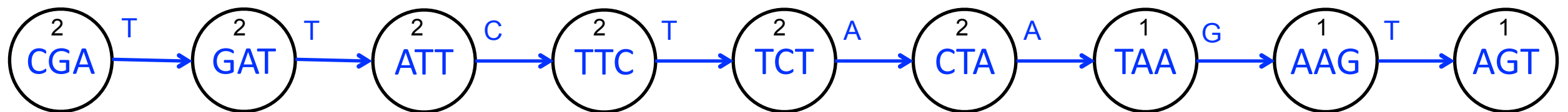
```
>seq3  
CGATTGTAAGT
```



Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

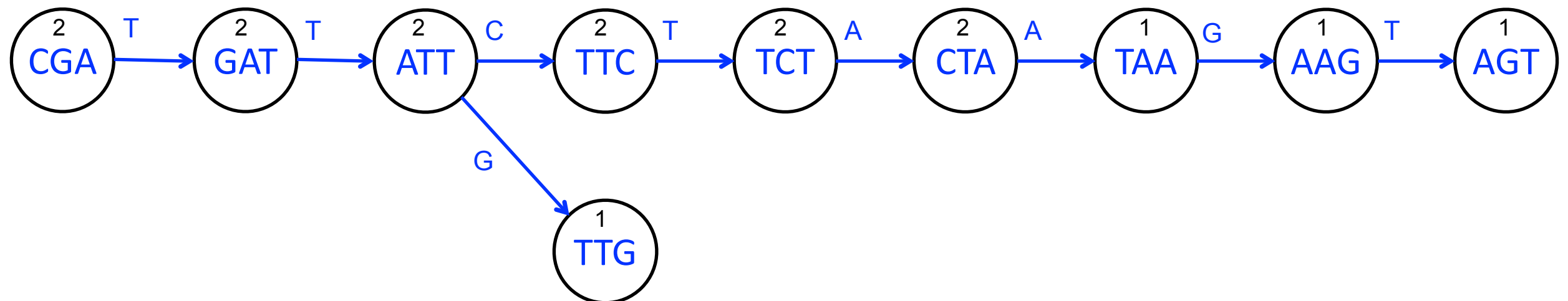
```
>seq3  
CGATTGTAAGT
```



Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

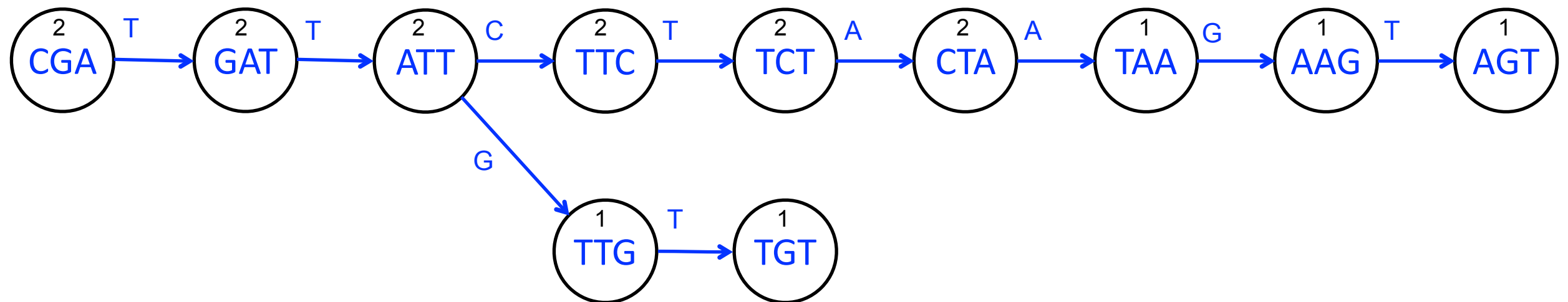
```
>seq3  
CGATTGTAAGT
```



Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

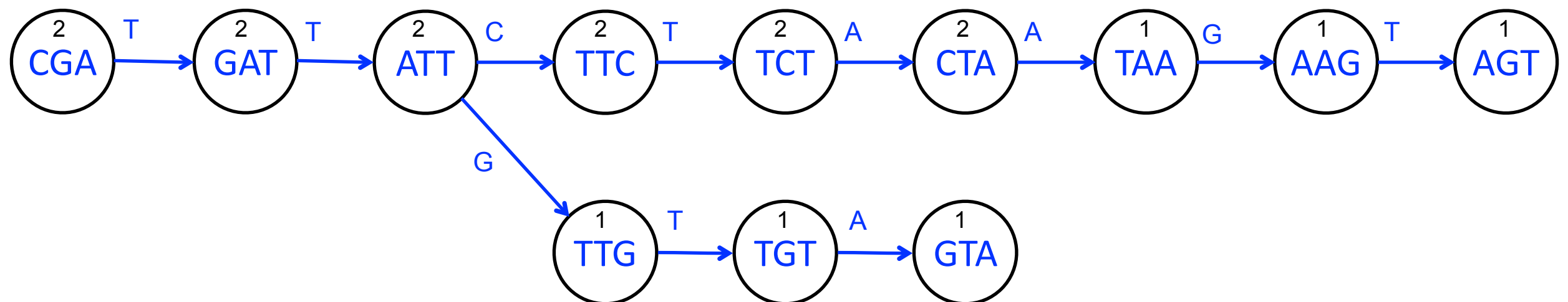
```
>seq3  
CGATTGTAAGT
```



Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

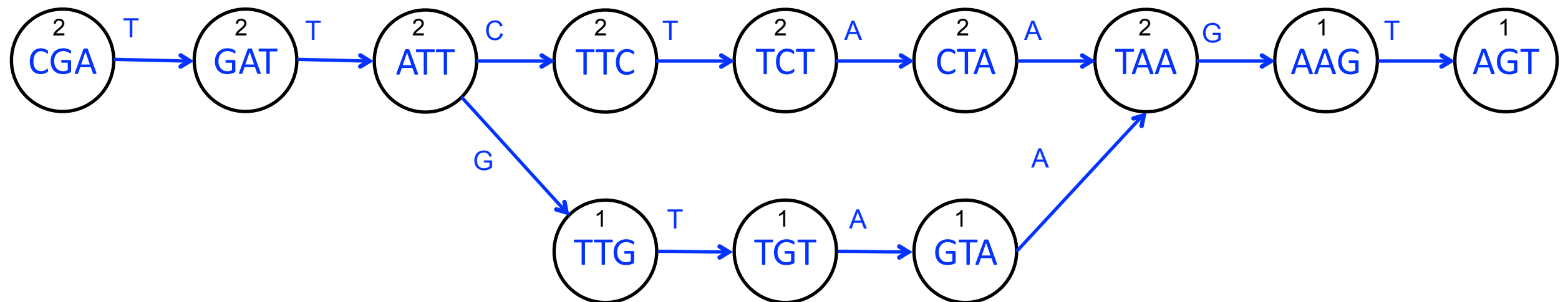
```
>seq3  
CGATTGTAAGT
```



Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

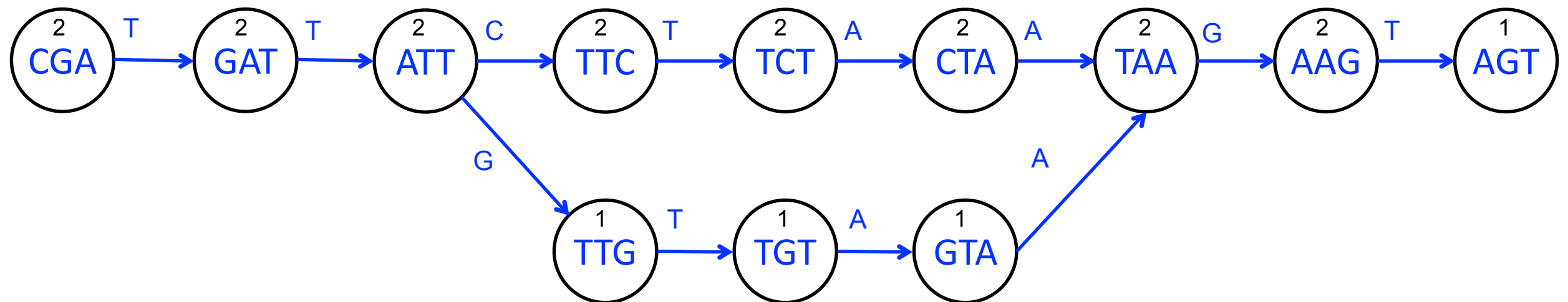
```
>seq3  
CGATTGTAAGT
```



Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

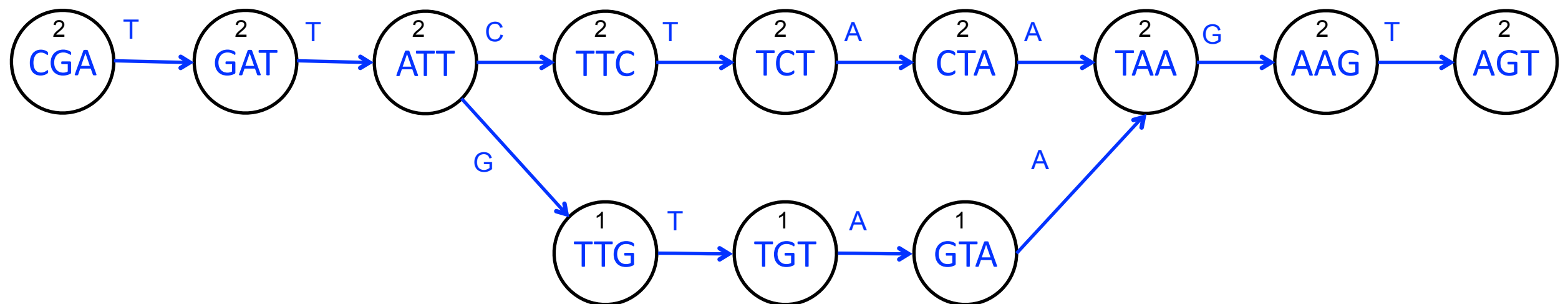
```
>seq3  
CGATTGTAAGT
```



Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

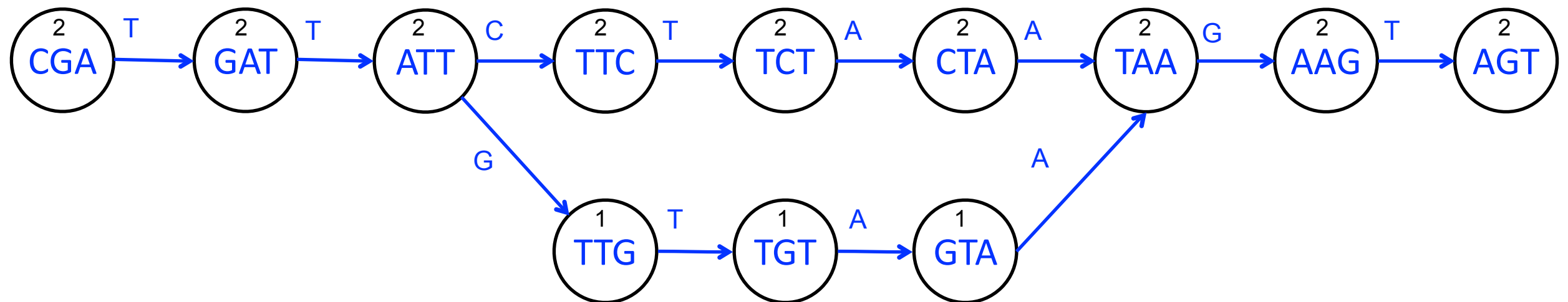
```
>seq3  
CGATTGTAAGT
```



Assembling a De Bruijn Graph

```
>seq1  
TTCTAAGT  
>seq2  
CGATTCTA
```

```
>seq3  
CGATTGTAAGT
```



CGATTCTAAGT
CGATTGTAAGT

Graphs get complicated

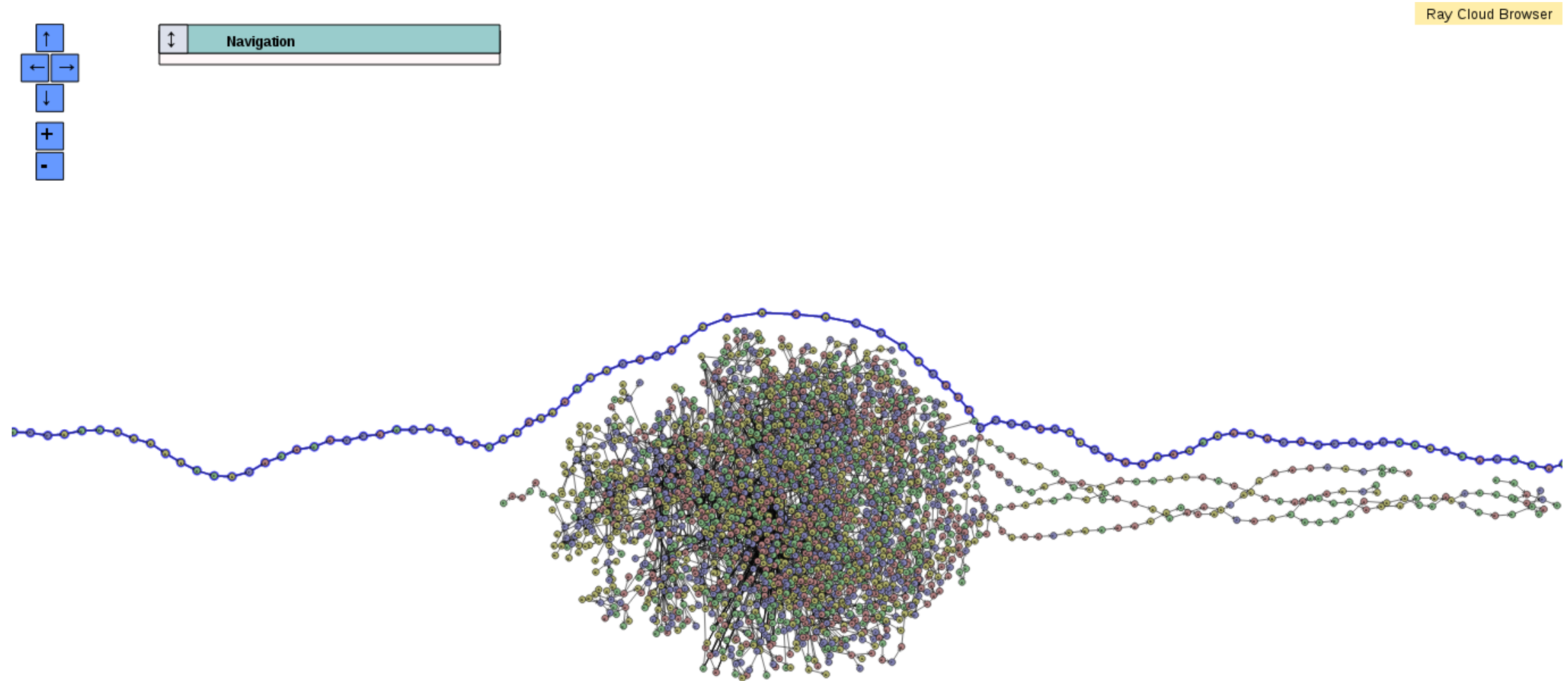
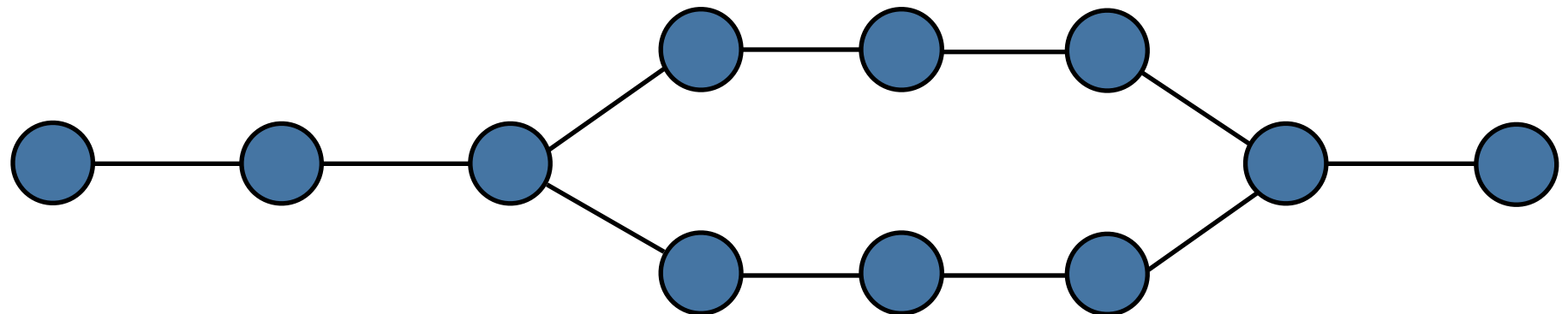


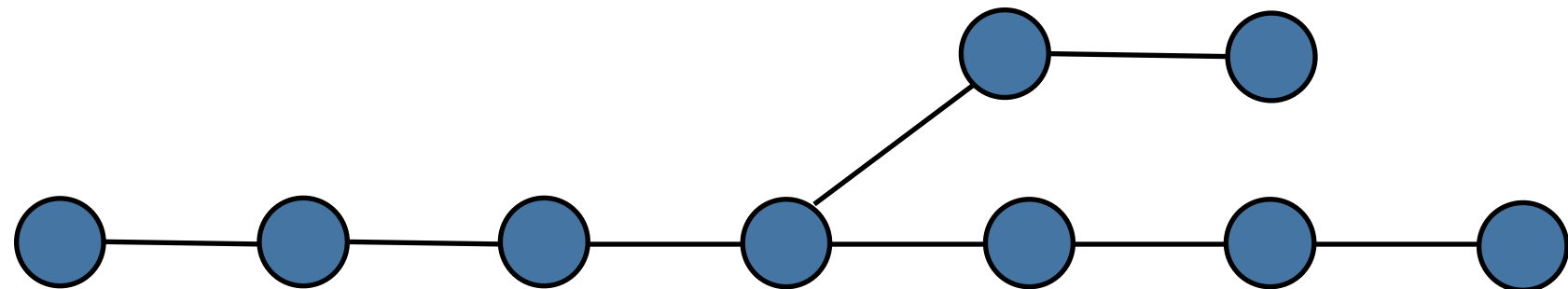
Image from dskernel.blogspot.co.uk

Common structures

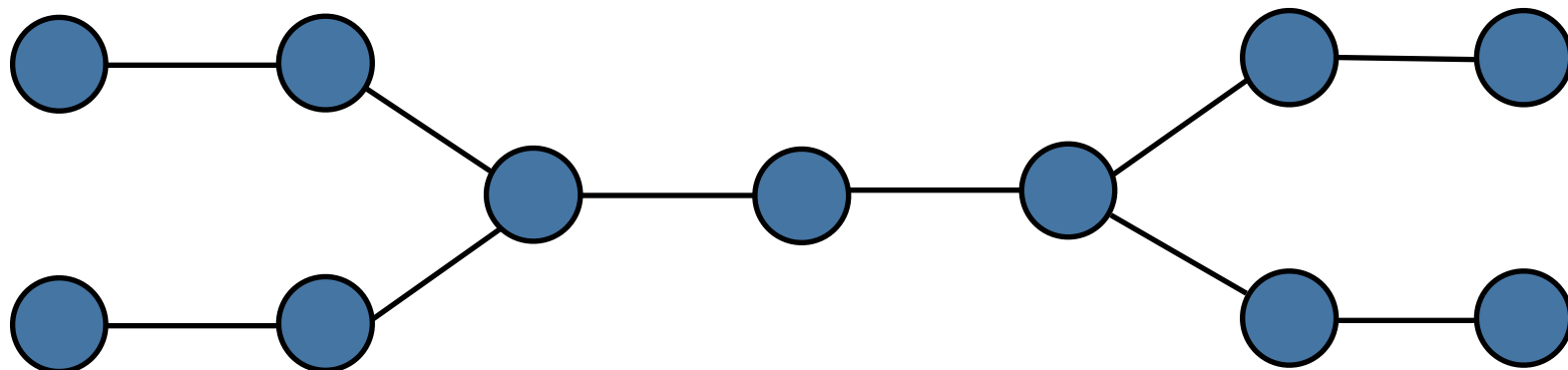
Polymorphism



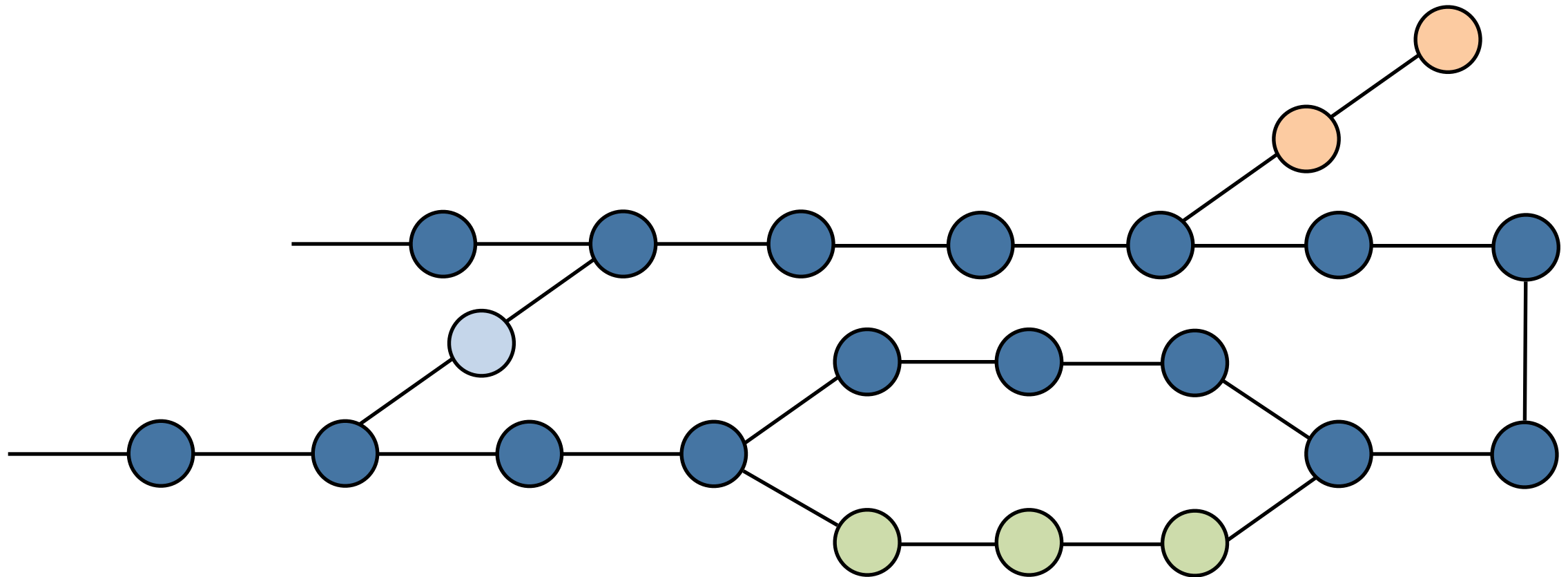
Errant base call



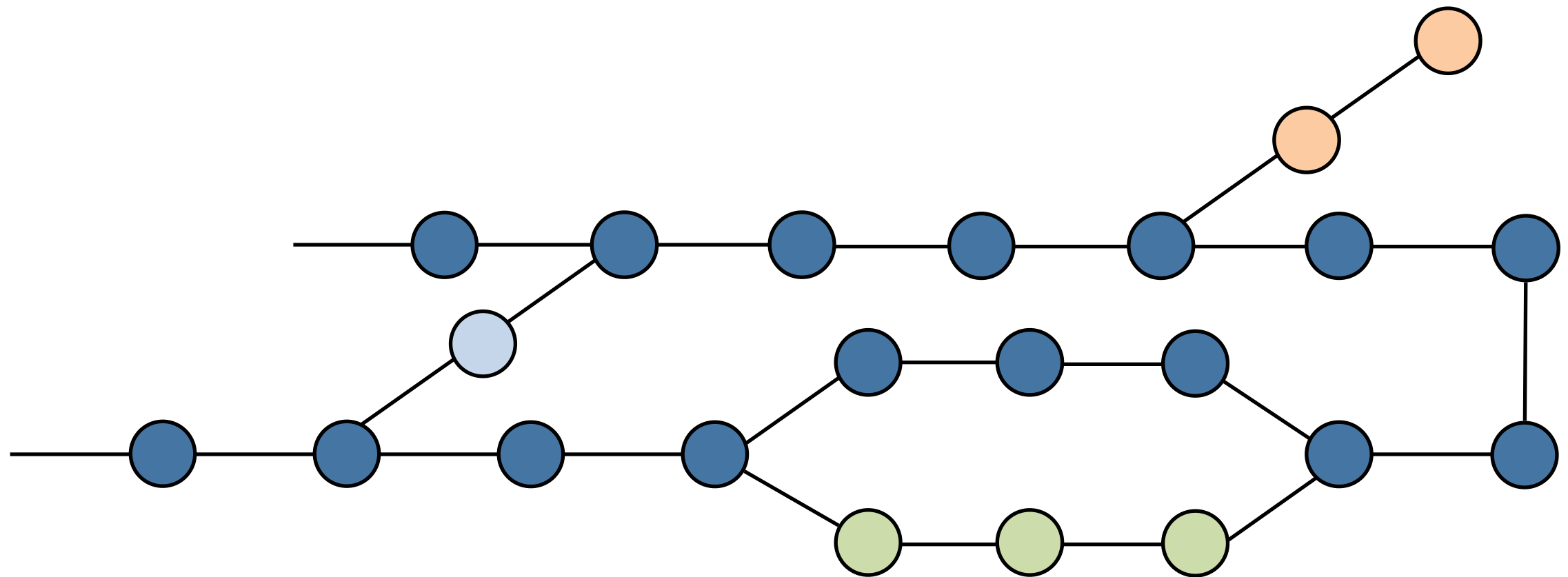
Repeating element



Cleaning graphs

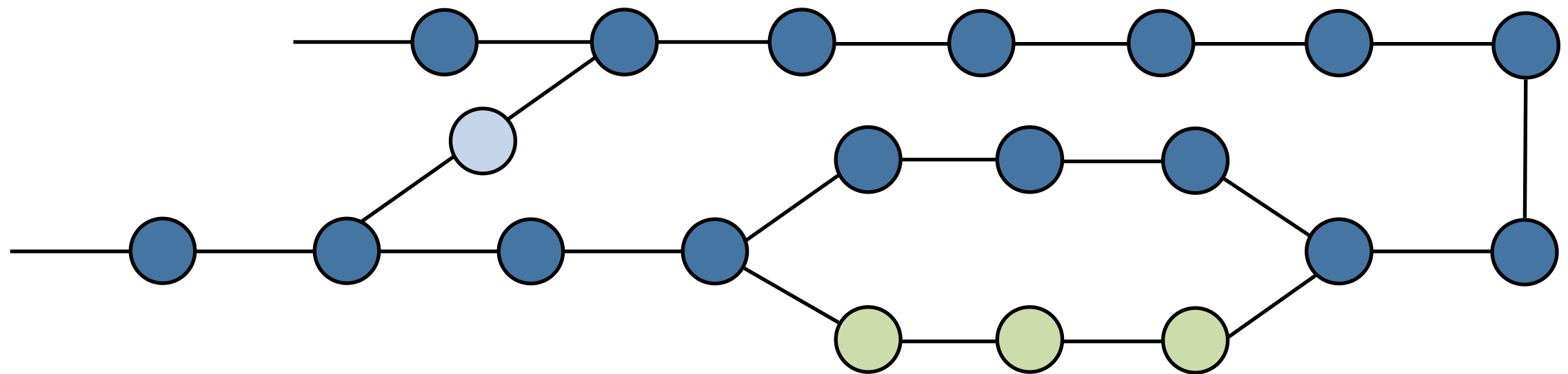



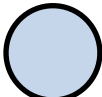
Cleaning graphs



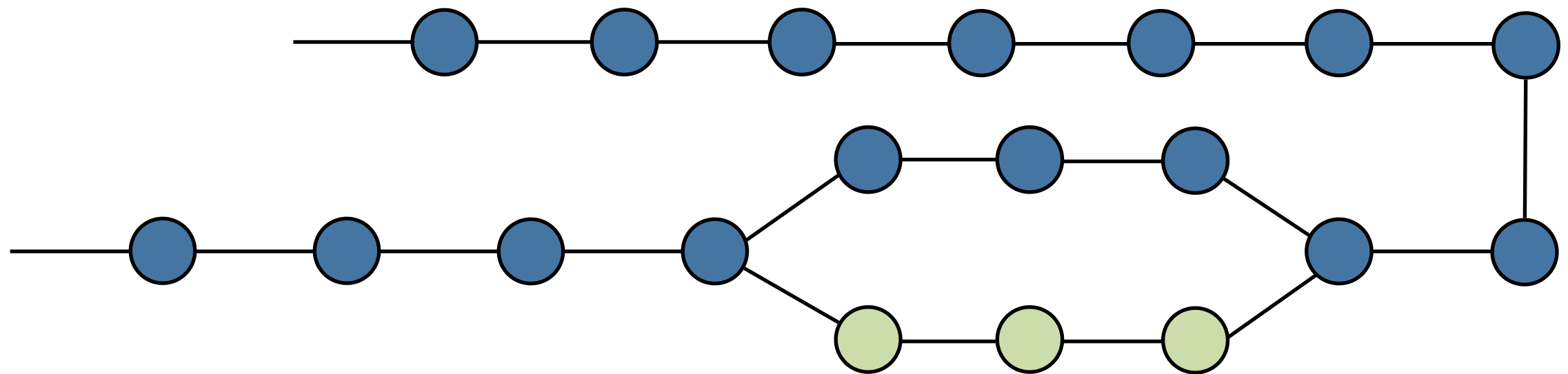
 Clip tips


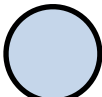

Cleaning graphs



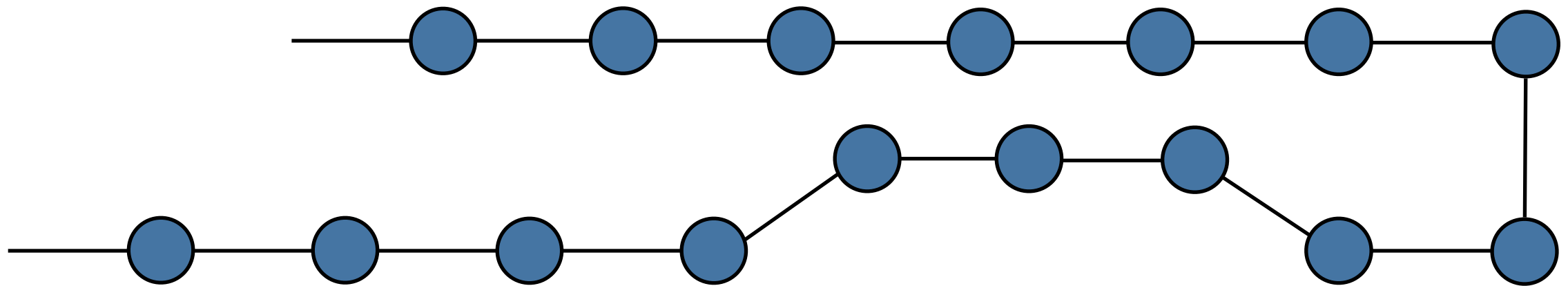
-  Clip tips
-  Remove low coverage nodes


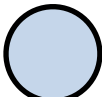

Cleaning graphs



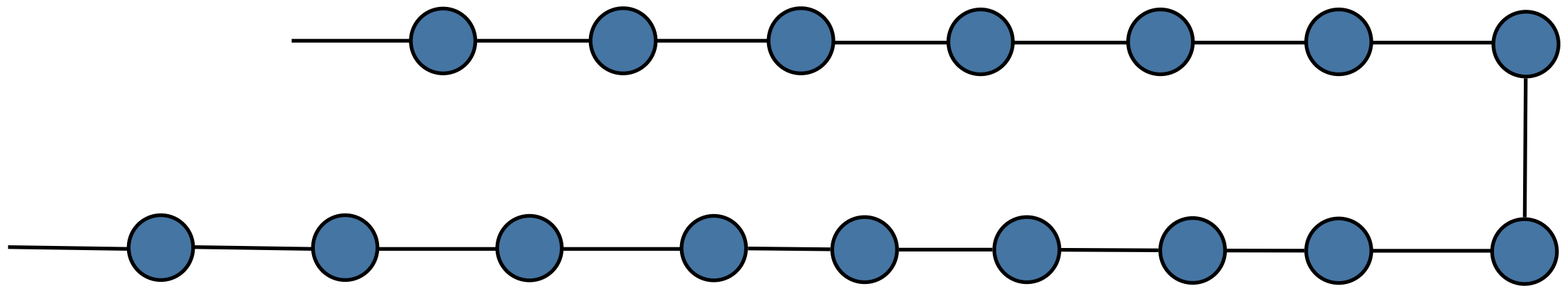
-  Clip tips
-  Remove low coverage nodes
-  Remove bubbles


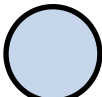

Cleaning graphs



-  Clip tips
-  Remove low coverage nodes
-  Remove bubbles

Cleaning graphs



-  Clip tips
-  Remove low coverage nodes
-  Remove bubbles

Resolving repeats using reads

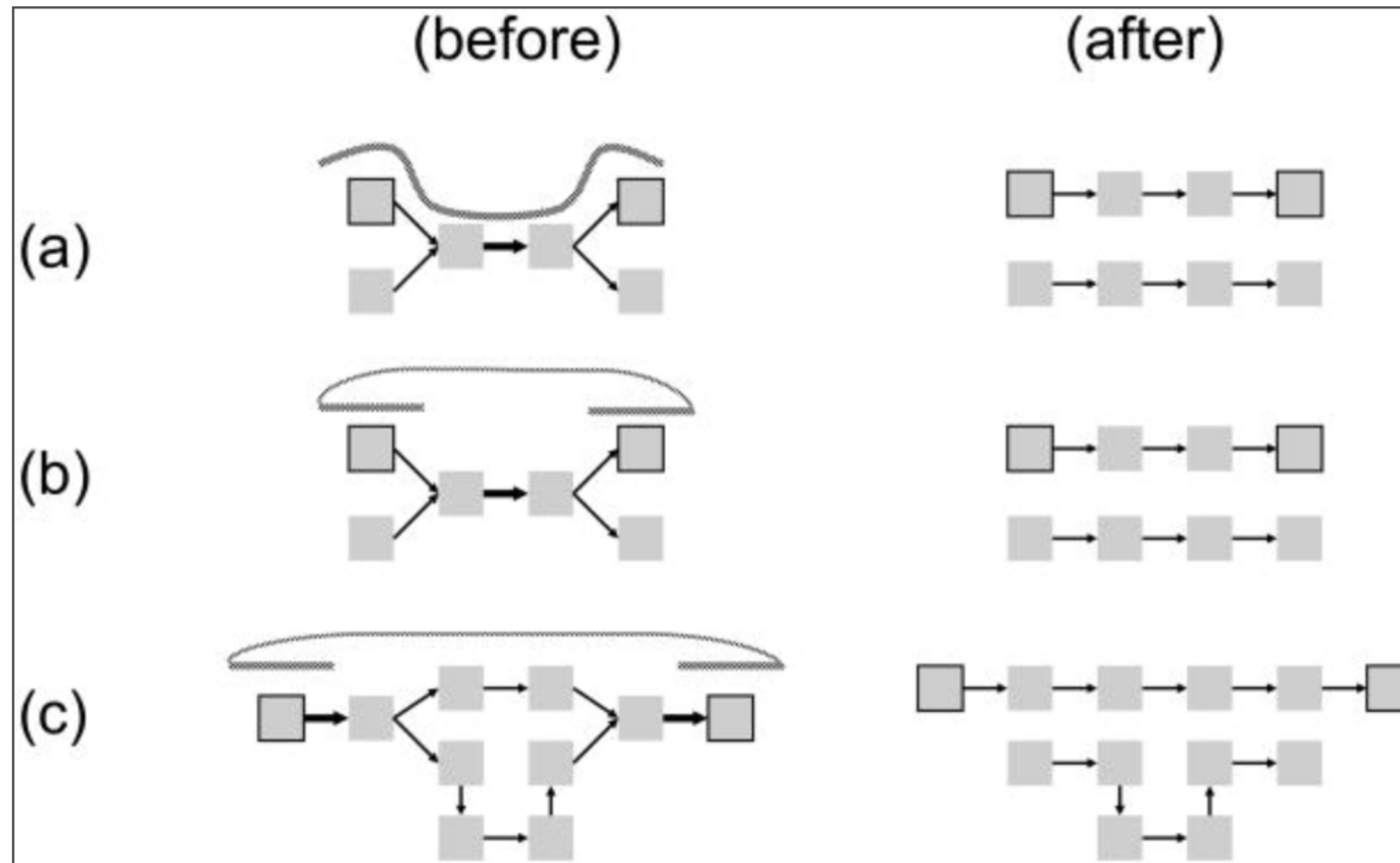


Figure 4

Three methods to resolve graph complexity. (a) Read threading joins paths across collapsed repeats that are shorter than the read lengths. (b) Mate threading joins paths across collapsed repeats that are shorter than the paired-end distances. (c) Path following chooses one path if its length fits the paired-end constraint. Reads and mates are shown as patterned lines. Not all tangles can be resolved by reads and mates. The non-branching paths are illustrative; they could be simplified to single edges or nodes.

[Assembly Algorithms for Next-Generation Sequencing Data](#)
Genomics. 2010 June;95(6):315-327.

OLC vs. De bruijn

	OLC	De bruijn
Representation of the problem	(Reads) Overlap Graph	De bruijn (kmer) graph
Steps to add a read	Insert read, compare to every read already included and insert overlaps	Insert new kmers or update count for those already present
Strengths	<ul style="list-style-type: none"> • Tracks reads • Intuitive representation • Consensus 	<ul style="list-style-type: none"> • Computational speed • Ability to handle big datasets
Optimal depth	Just enough to cover genome and give accurate consensus	The higher the better (to grow SNR)
Typical sequencing technologies processed	Sanger, 454, Pacbio	Illumina, Ion Torrent

Example datasets

- Dataset A:
 - Target size ~5Mbp.
 - Illumina 2x300bp.
 - Pacbio long reads.
- Dataset B:
 - Target Size ~12Mbp.
 - Illumina 2x100bp.
 - Nextera LMP (2x300bp).

```
dataset_A
dataset_A/pacbio_long.fastq
dataset_A/pe_1.fastq
dataset_A/pe_2.fastq
dataset_B
dataset_B/nextera_lmp_1.fastq
dataset_B/nextera_lmp_2.fastq
dataset_B/pe_1.fastq
dataset_B/pe_2.fastq
```

Questions?

