

# PAIRED DE BRUIJN GRAPHS

**22BIO211: Intelligence of  
Biological Systems - 2**

Dr. Manjusha Nair M  
Amrita School of Computing, Amritapuri  
Email : [manjushanair@am.amrita.edu](mailto:manjushanair@am.amrita.edu)  
Contact No: 9447745519

# Constructing Paired de Bruijn graphs from Genome

---

1. Representing Genome as a Path
2. Labeling Nodes by Paired Prefixes and Suffixes
3. Glue nodes with identical labels

# Constructing Paired de Bruijn graphs from paired k-mers

**Paired de Bruijn graph for a collection of paired  $k$ -mers:**

- Represent every paired  $k$ -mer as an edge between its paired prefix and paired suffix.
- Glue **ALL** nodes with identical labels.

## Prefix – Suffix of (k,d) -mers

- Given a (k, d)-mer ( $a_1 \dots a_k \mid b_1, \dots, b_k$ ), we define its **prefix** and **suffix** as the following ( $k-1, d+1$ ): mers

$$\text{PREFIX}((a_1 \dots a_k \mid b_1, \dots, b_k)) = (a_1 \dots a_{k-1} \mid b_1 \dots b_{k-1})$$

$$\text{SUFFIX}((a_1 \dots a_k \mid b_1, \dots, b_k)) = (a_2 \dots a_k \mid b_2 \dots b_k)$$

For example,

- $\text{PREFIX}((GAC \mid TCA)) = (GA \mid TC)$
- $\text{SUFFIX}((GAC \mid TCA)) = (AC \mid CA)$ .

## Prefix – Suffix of (k,d) -mers

- For consecutive (k, d)-mers appearing in Text, the suffix of the first (k, d)-mer is equal to the prefix of the second (k, d)-mer.

For example, for the consecutive (k, d)-mers  
 $(TAA \mid GCC)$  and  $(AAT \mid CCA)$  in  
**TAA**TCATGGGATGTT**,**

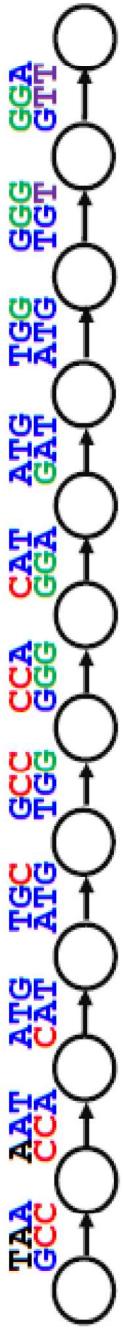
$$\text{SUFFIX}((TAA \mid GCC)) = \text{PREFIX}((AAT \mid CCA)) = (AA \mid CC).$$

# 1. Representing Genome as a Path Graph

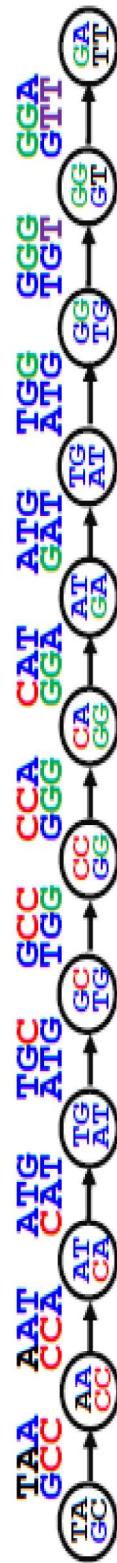


PathGraph : edges corresponding to all  $(k, d)$ -mers.

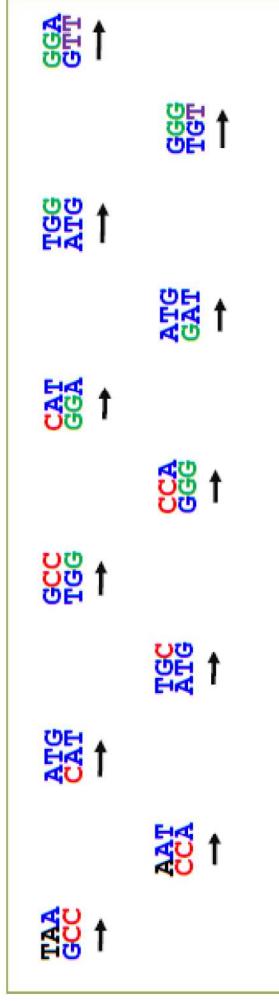
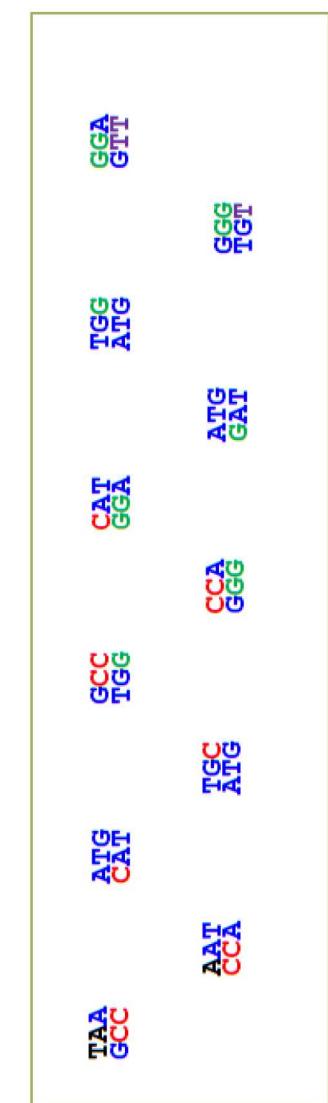
## 2. Labeling Nodes by Paired Prefixes and Suffixes



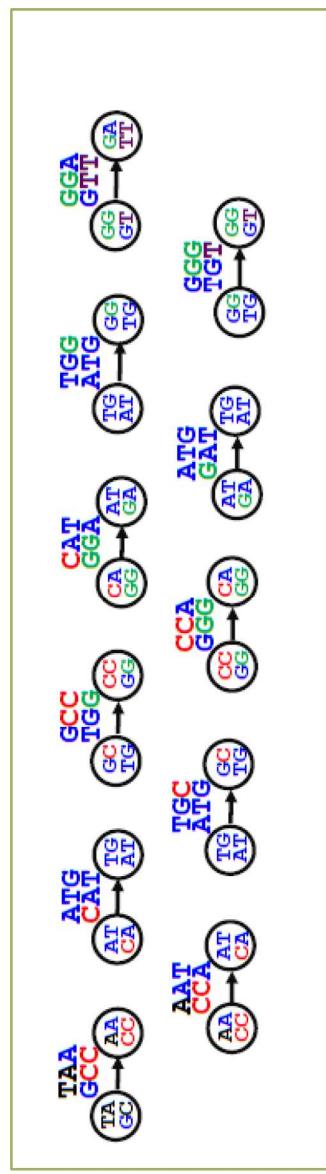
Labeling Nodes by Paired Prefixes and Suffixes



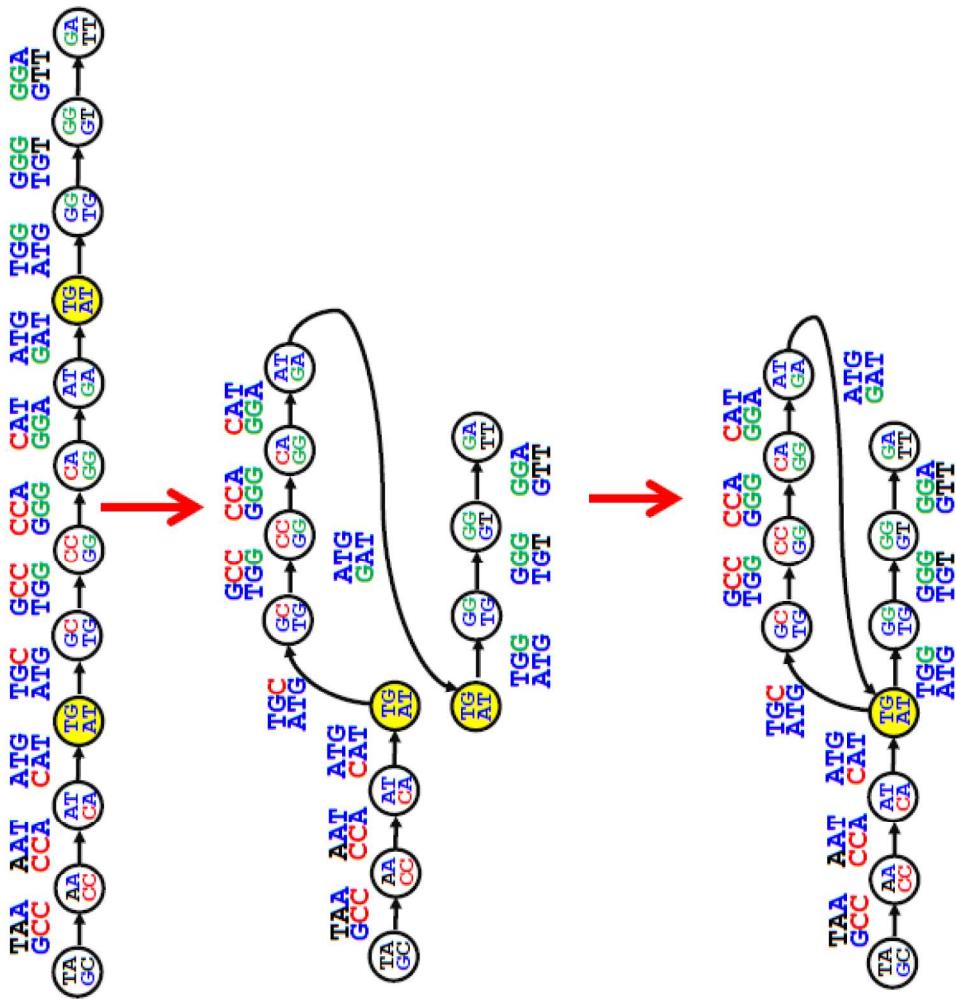
paired prefix of  $\begin{matrix} \text{CCA} \\ \text{GGG} \end{matrix}$   $\rightarrow$   $\begin{matrix} \text{CC} \\ \text{GG} \end{matrix}$   $\leftarrow$  paired suffix of  $\begin{matrix} \text{CCA} \\ \text{GGG} \end{matrix}$

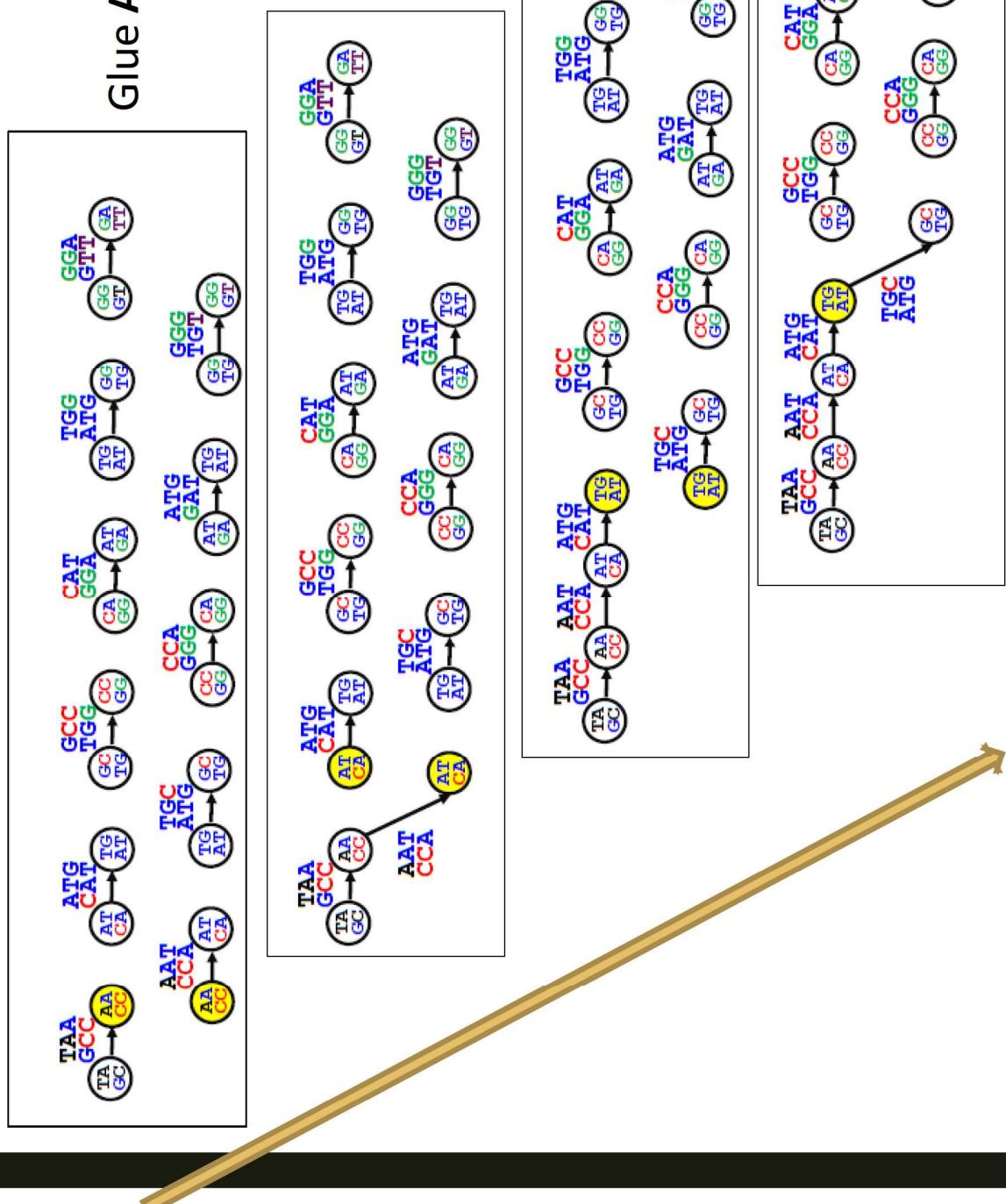


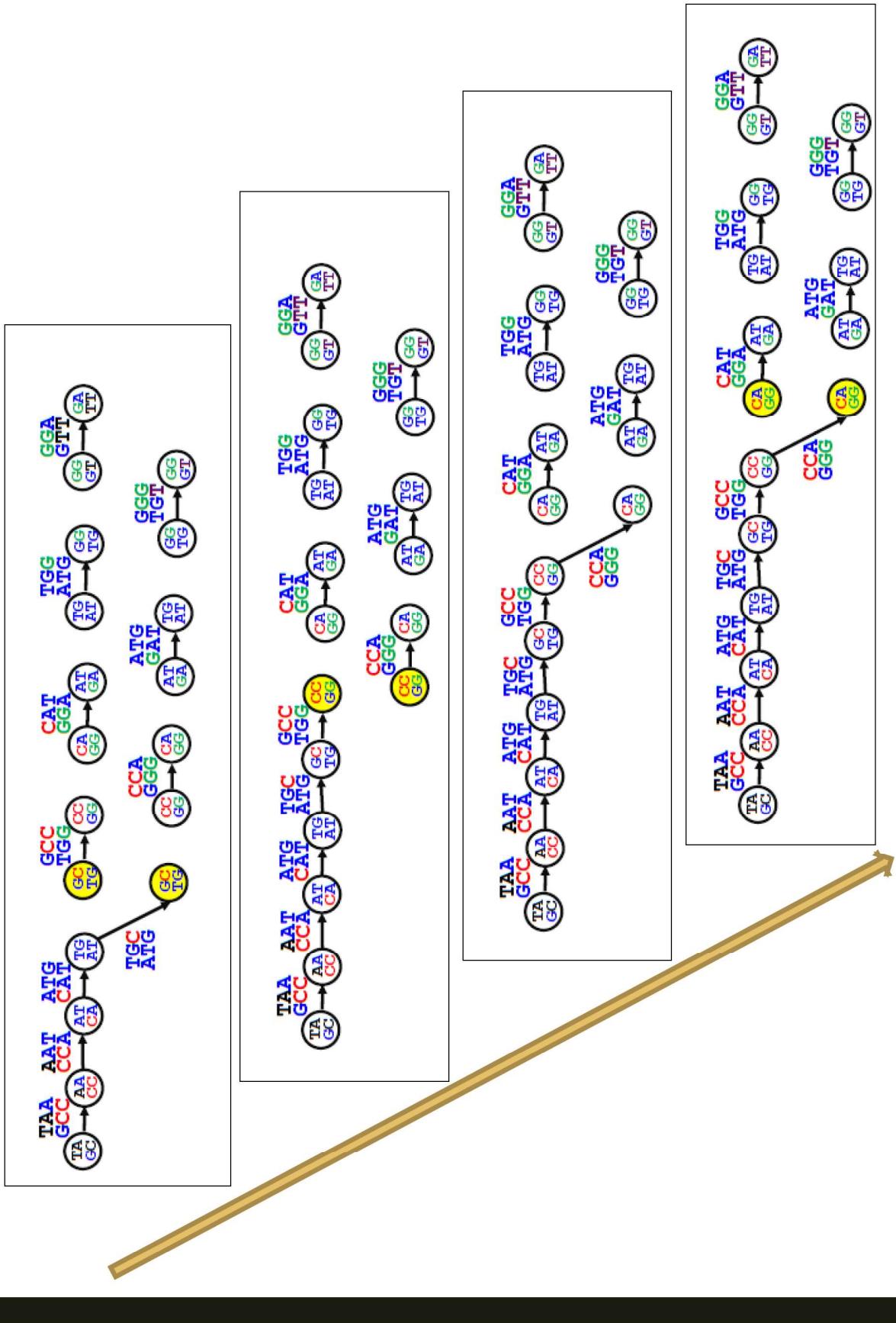
paired prefix of **CCA** → **GGG** ← paired suffix of **CCA** **GGG**

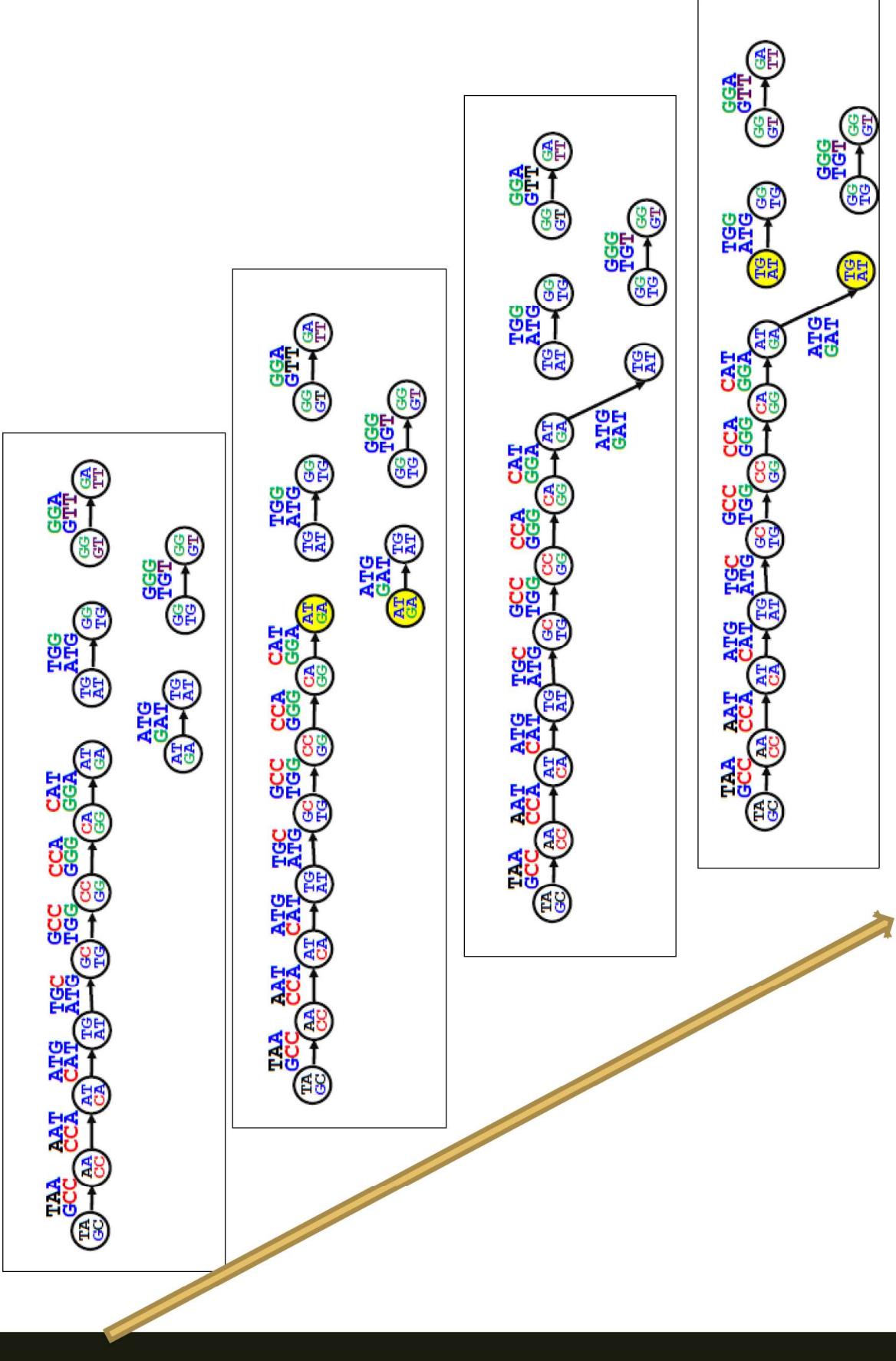


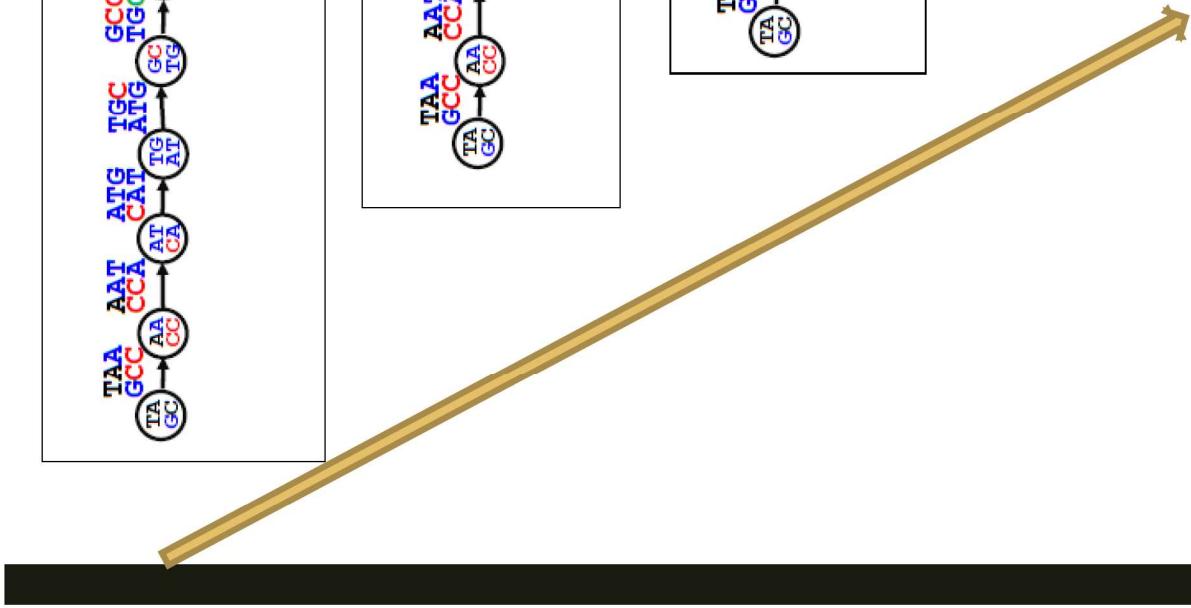
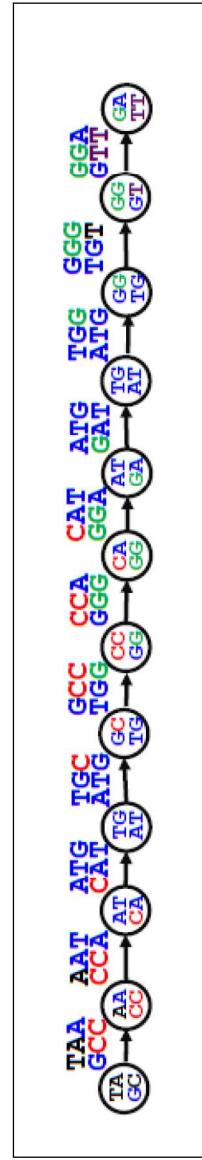
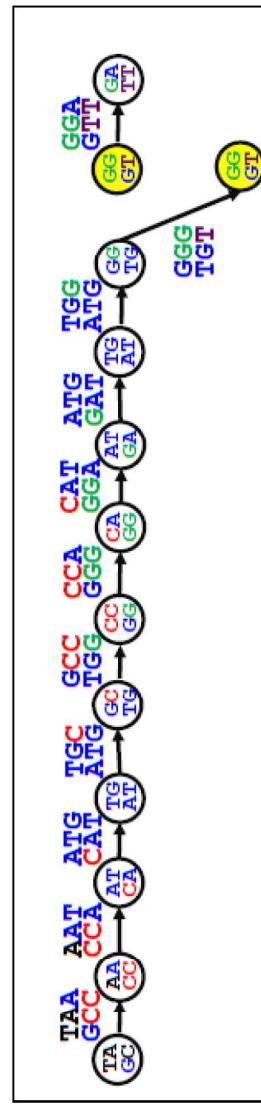
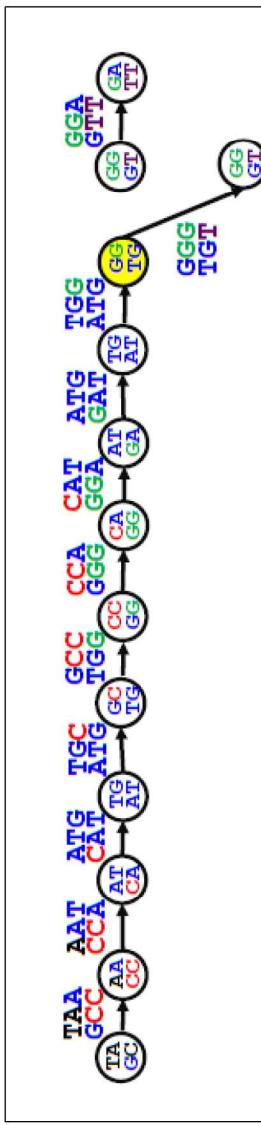
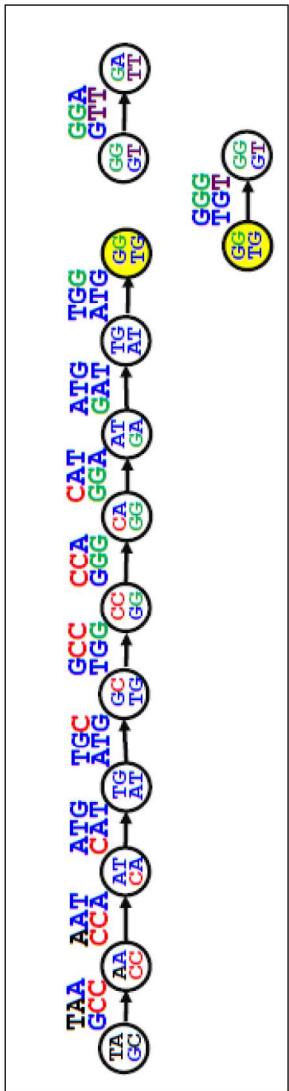
### 3. Glue nodes with identical labels



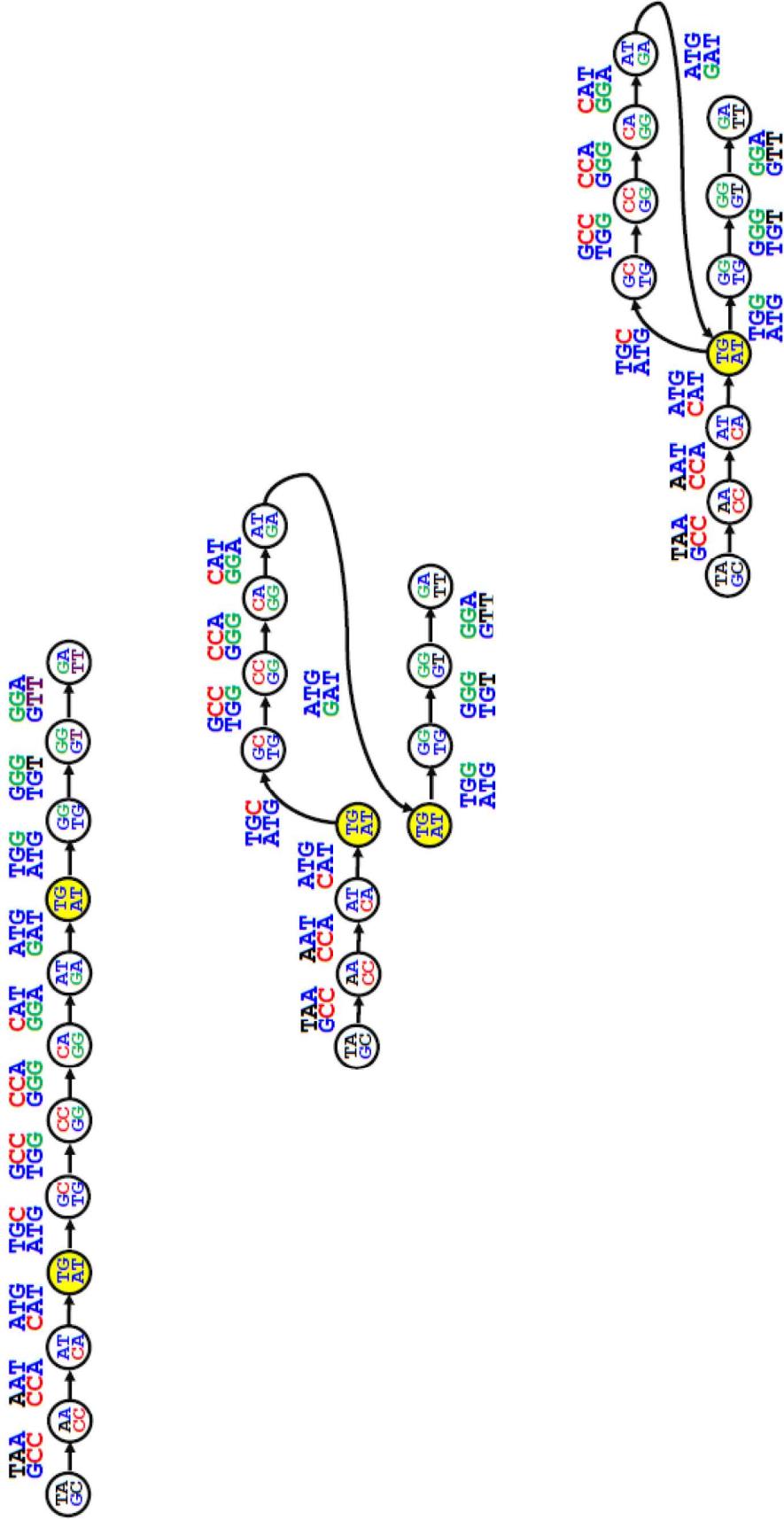






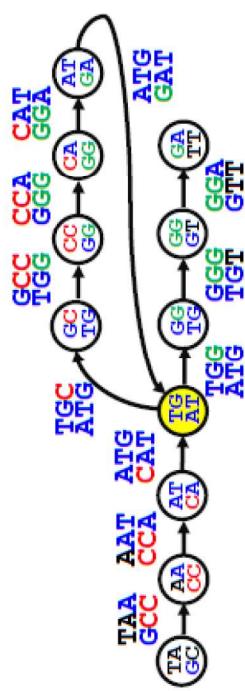


## We Are Not Done with Gluing Yet

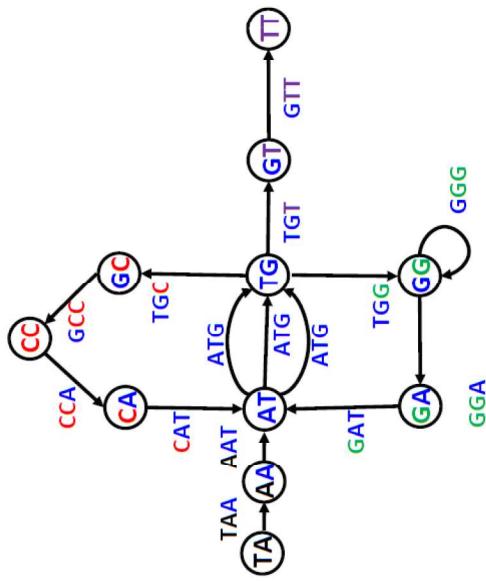


# Which Graph Represents a Better Assembly?

Paired de Bruijn Graph



de Bruijn Graph



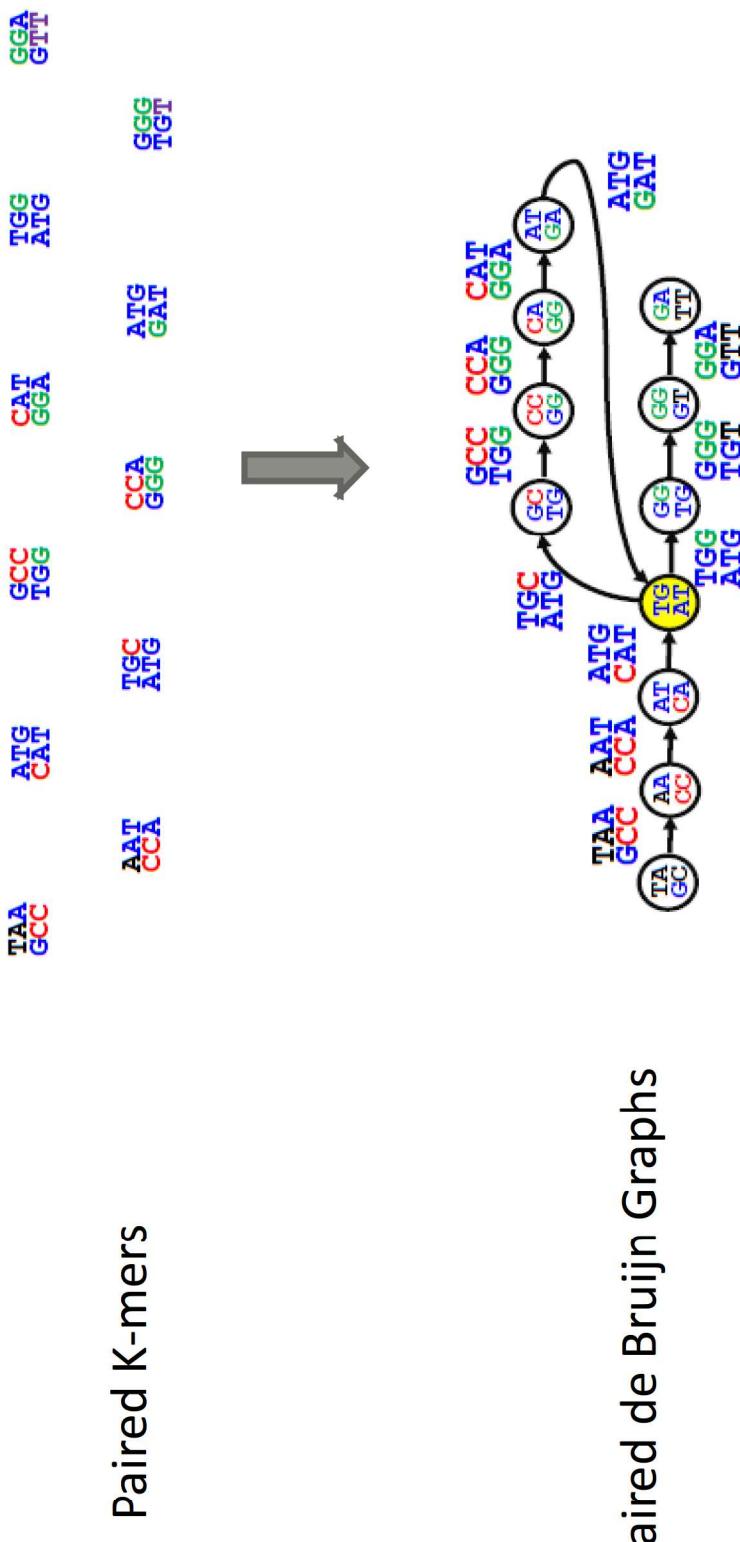
Unique genome reconstruction

**TAATGCCATGGATGTT**

Multiple genome reconstructions

**TAATGGATGCCATGTT**

# Paired deBrujin Graphs



# String Reconstruction

---

## Solution of the String Reconstruction Problem:

- Eulerian path in the de Bruijn graph constructed from a  $k$ -mer composition.

## Solution of the String Reconstruction Problem from Read Pairs:

- Eulerian path in the de Bruijn graph constructed from a  $(k,d)$ -mer composition.

# Eulerian Path through Paired deBrujin Graphs

---

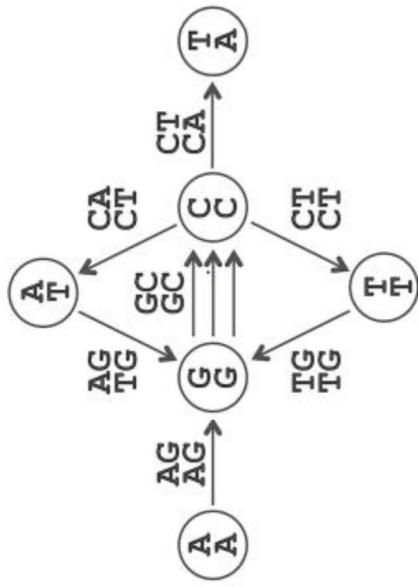
We saw that every Eulerian path in the de Brujin graph constructed from a k-mer composition spells out a solution of the String Reconstruction Problem.

But is this the case for the paired de Brujin graph?

# String Reconstruction using Eulerian path in paired de Bruijn graphs

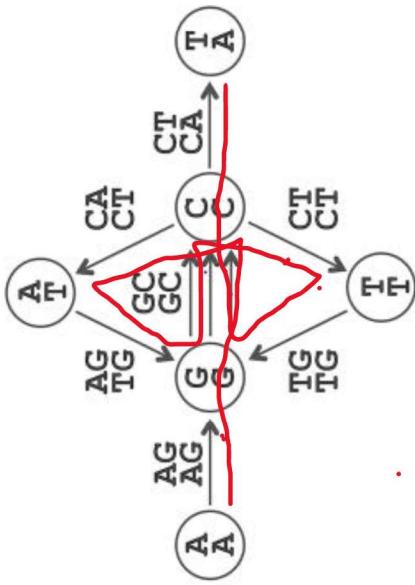
Create de Bruijn Graph: constructed from the collection of nine (2,1)-mers

(AG|AG), (AG|TG), (CA|CT), (CT|CA), (CT|CT), (GC|GC), (GC|GC), (TG|TG).



# String Reconstruction using Eulerian path in paired de Bruijn graphs

Reconstruct the genome spelled by the following Eulerian path

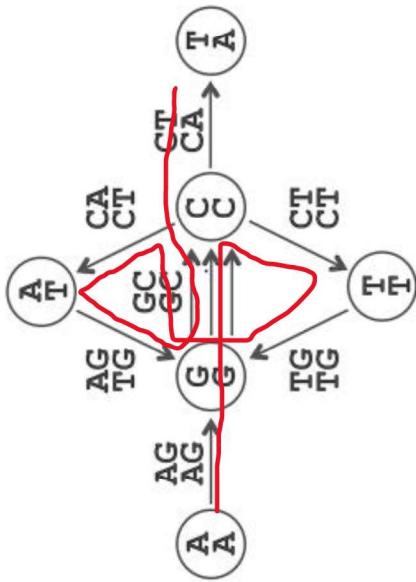


(AG|AG) → (GC|GC) → (CA|CT) → (AG|TG) → (GC|GC) →  
(CT|CT) → (TG|TG) → (GC|GC) → (CT|CA)

AG-AG  
GC-GC  
CA-CT  
AG-TG  
GC-GC  
CT-CT  
TG-TG  
GC-GC  
CT-CA  
**AGCAGCTGCTGCA**

# A pitfall of paired de Bruijn graphs

- Look again at the same graph, and consider the following Eulerian Path:
  - $(AG|AG) \rightarrow (GC|GC) \rightarrow (CT|CT) \rightarrow (TG|TG) \rightarrow (GC|GC) \rightarrow (CA|CT) \rightarrow (AG|TG) \rightarrow (GC|GC) \rightarrow (CT|CA)$ .



**AG-~~AG~~  
GC-GC  
CT-CT  
TG-TG  
CA-CT  
GC-GC  
AG-TG  
GC-GC  
CT-CA**

Can you reconstruct a genome spelled by this path?

AGC?GC?GCTGCA

Here, not every column has the same nucleotide

# A pitfall of paired de Bruijn graphs

Not every Eulerian path in the paired de Bruijn graph constructed from  $(k,d)$  -mer composition spells out a solution of the String Reconstruction from Read-Pairs Problem.

# String Spelled by a Gapped Genome Path Problem

---

**String Spelled by a Gapped Genome Path Problem:**

*Reconstruct a sequence of  $(k, d)$ -mers corresponding to a path in a paired de Bruijn graph.*

**Input:** A sequence of  $(k, d)$ -mers  $(a_1|b_1), \dots, (a_n|b_n)$  such that  
 $\text{SUFFIX}((a_i|b_i)) = \text{PREFIX}((a_{i+1}|b_{i+1}))$  for  $1 \leq i \leq n - 1$ .

**Output:** A string *Text* of length  $k + d + k + n - 1$  such that the  $i$ -th  $(k, d)$ -mer  
of *Text* is equal to  $(a_i|b_i)$  for  $1 \leq i \leq n$  (if such a string exists).

---

# String Spelled by a Gapped Genome Path Problem

---

Our approach

1. Split the given  $(k, d)$ -mers  $(a_1 | b_1), \dots, (a_n | b_n)$  into their initial  $k$ -mers,  $FirstPatterns = (a_1, \dots, a_n)$ , and their terminal  $k$ -mers,  $SecondPatterns = (b_1, \dots, b_n)$ .
2. Assemble  $FirstPatterns$  and  $SecondPatterns$  into strings  $PrefixString$  and  $SuffixString$
3. Concatenate the two strings

# String Spelled by a Gapped Genome Path Problem: Examples

AG-**A**G  
GC-GC  
**CA**-CT  
AG-TG  
GC-GC  
**CT**-CT  
TG-TG  
GC-GC  
**CA**-CT  
AG-TG  
GC-GC  
**CT**-CA

PrefixString = AGCTGGCAGCT

SuffixString = AGCTGCTGCA

There is no perfect overlap

PrefixString = AGCTGCA**G**CT  
SuffixString = **A**GCTGCTGCA  
Genome = AGC?**G**C?GCTGCA

PrefixString = AGCAGCTGCT  
SuffixString = AGCTGCTGCA  
Genome = AGCAGCTGCTGCA

# Summary

- String Reconstruction from Read-Pairs
- Paired de Bruijn graphs
- Constructing Paired de Bruijn graphs from Genome
- Constructing Paired de Bruijn graphs from paired k-mers
- Pitfall of Paired de Bruijn graphs
- Eulerian path through Paired de Bruijn graphs
- String Spelled by a Gapped Genome Path Problem