

Computational Biology

DNA Sequencing

Department of Mathematics
Stockholm University

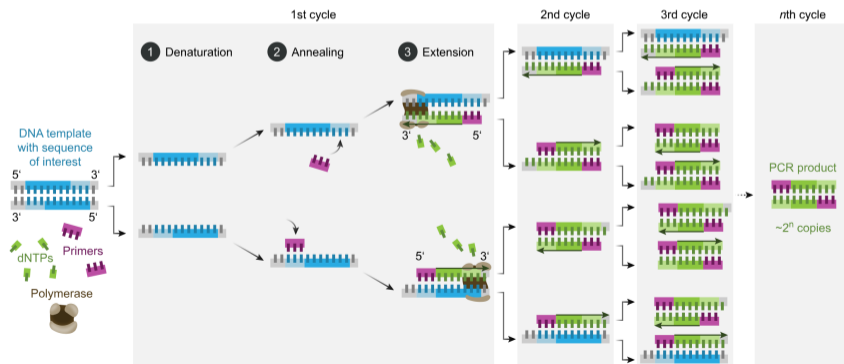
- ▶ Copying DNA:
 - ▶ Polymerase chain reaction (PCR)
- ▶ Sequencing DNA:
 - ▶ Sanger Sequencing [AKA 1st generation sequencing]
 - ▶ Next/2nd-generation sequencing (NGS) [AKA Massive parallel sequencing]
 - ▶ 3rd-generation [AKA long-read sequencing]

Polymerase chain reaction (PCR)

- ▶ used to copy DNA
- ▶ Invented by Kary Mullis (Nobel prize 1993)
- ▶ **Input:** a DNA "template" t to copy, primers, polymerase, bases A, C, G, T ,
Process: n "cycles" (see right)
Output: roughly 2^n copies of t

Per cycle there are 3 phases:

- 1 Denature: 94-98 °C for 20–30 s
- 2 Anneal: 50-65 °C for 20–40 s
- 3 Extension: 75-80 °C

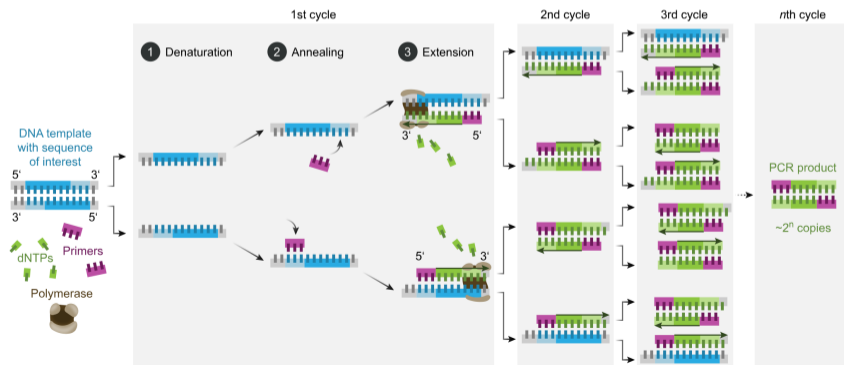


Polymerase chain reaction (PCR)

- ▶ used to copy DNA
- ▶ Invented by Kary Mullis (Nobel prize 1993)
- ▶ **Input:** a DNA "template" t to copy, primers, polymerase, bases A, C, G, T ,
Process: n "cycles" (see right)
Output: roughly 2^n copies of t

Per cycle there are 3 phases:

- 1 Denature: 94-98 °C for 20–30 s
- 2 Anneal: 50-65 °C for 20–40 s
- 3 Extension: 75-80 °C



Sanger Sequencing

- ▶ used to read "small (500bp)" DNA sequences
- ▶ Invented by Fredrick Sanger and coworkers, 1977 (Nobel prize 1980)
- ▶ **Input:** copies of DNA split into 4 test tubes that contains primers, polmerase, bases, "modified bases A, C, T, G "
Each tube contains all bases and ONE "modified base"
 $I \in \{A, C, G, T\}$

Process (Basic Idea): "modified base" I ensures that when added during reading process of one DNA-copy, the reading process stops.

Having multiple copies and the four tubes, this ensures, that (with high probability) the tupe I contains all single strands that end with I .

gel electrophoresis: reads are negative charged and small reads get "closer" to positive pol (proportional to their length)

Output: the read of the input DNA

Sanger Sequencing

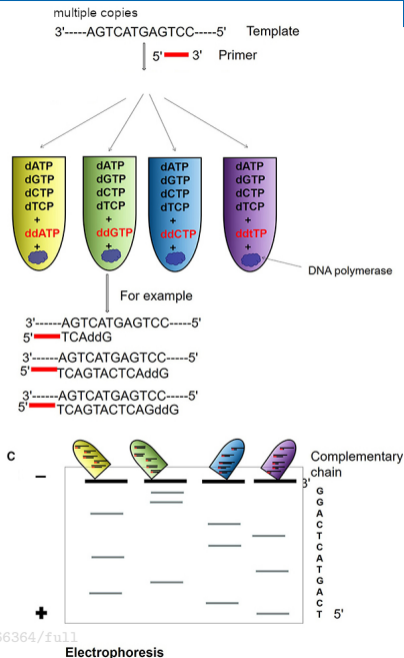
- ▶ used to read "small (500bp)" DNA sequences
- ▶ Invented by Fredrick Sanger and coworkers, 1977 (Nobel prize 1980)
- ▶ **Input:** copies of DNA split into 4 test tubes that contains primers, polmerase, bases, "modified bases A, C, T, G"
Each tube contains all bases and ONE "modified base"
 $I \in \{A, C, G, T\}$

Process (Basic Idea): "modified base" I ensures that when added during reading process of one DNA-copy, the reading process stops.

Having multiple copies and the four tubes, this ensures, that (with high probability) the tupe I contains all single strands that end with I .

gel electrophoresis: reads are negative charged and small reads get "closer" to positive pol (proportional to their length)

Output: the read of the input DNA



Sanger Sequencing

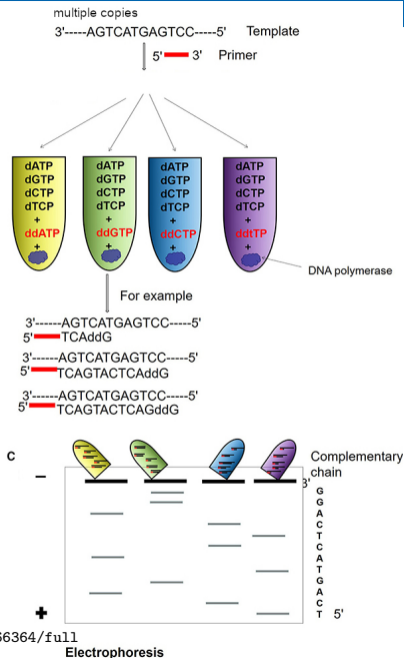
- ▶ used to read "small (500bp)" DNA sequences
- ▶ Invented by Fredrick Sanger and coworkers, 1977 (Nobel prize 1980)
- ▶ **Input:** copies of DNA split into 4 test tubes that contains primers, polmerase, bases, "modified bases A, C, T, G"
Each tube contains all bases and ONE "modified base"
 $I \in \{A, C, G, T\}$

Process (Basic Idea): "modified base" I ensures that when added during reading process of one DNA-copy, the reading process stops.

Having multiple copies and the four tubes, this ensures, that (with high probability) the tupe I contains all single strands that end with I .

gel electrophoresis: reads are negative charged and small reads get "closer" to positive pol (proportional to their length)

Output: the read of the input DNA



Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

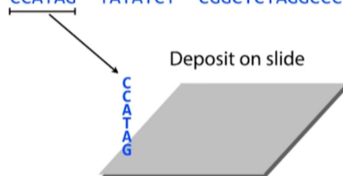
Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

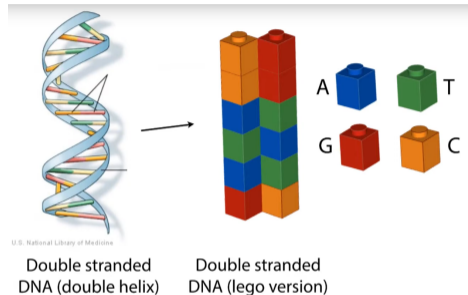
Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

```
CCATAGTA TATCTCGG CTCTAGGCCCTC ATTTTTT  
CCA TAGTATAT CTCGGCTCTAGGCCCTCA TTTTTT  
CCATAGTAT ATCTCGGCTCTAG GCCCTCA TTTTTT  
CCATAG TATATCT CGGCTCTAGGCCCT CATTTTTT
```



Next-generation sequencing (NGS)

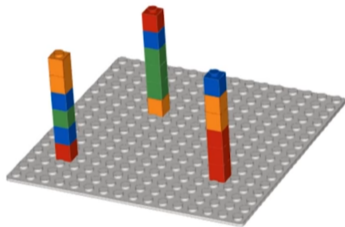
- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":
Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..



Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

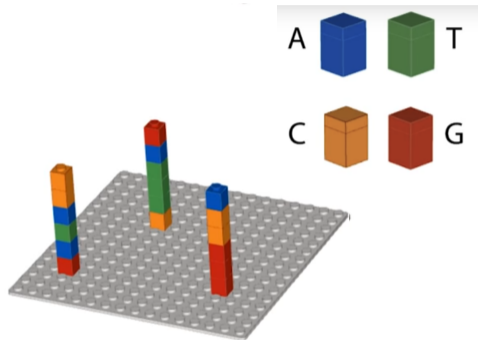
Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..



Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

Input: copies of **multiple** DNA (fragments) placed on a slide, bases, **terminators**, polymerase, ..

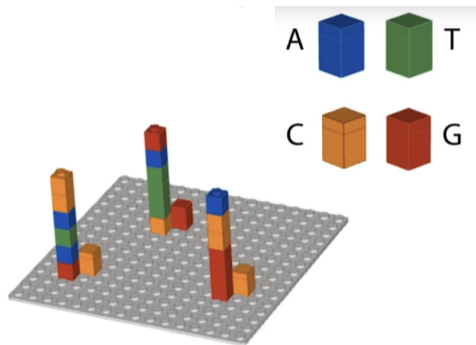


Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

Input: copies of **multiple** DNA (fragments) placed on a slide, bases, **terminators**, polymerase, ..

Process (Basic Idea): when bases with terminators bind, no further base can be added.



Next-generation sequencing (NGS)

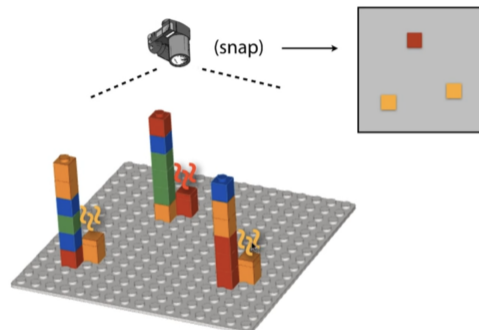
- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

Process (Basic Idea): when bases with terminators bind, no further base can be added.

terminators are engineered to glow a particular color (A, C, G, T)

→ take photo



Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exist, one is the "Illumina sequencing process":

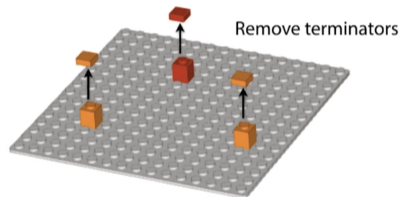
Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

Process (Basic Idea): when bases with terminators bind, no further base can be added.

terminators are engineered to glow a particular color (A, C, G, T)

→ take photo

after taking photo, terminators are removed and process is repeated.



Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

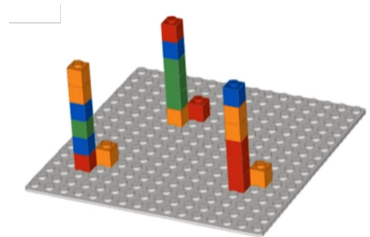
Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

Process (Basic Idea): when bases with terminators bind, no further base can be added.

terminators are engineered to glow a particular color (A, C, G, T)

→ take photo

after taking photo, terminators are removed and process is repeated.



Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

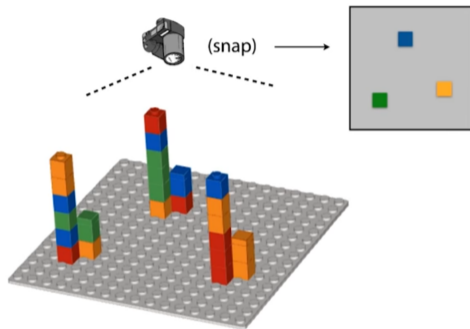
Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

Process (Basic Idea): when bases with terminators bind, no further base can be added.

terminators are engineered to glow a particular color (*A, C, G, T*)

→ take photo

after taking photo, terminators are removed and process is repeated.



Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

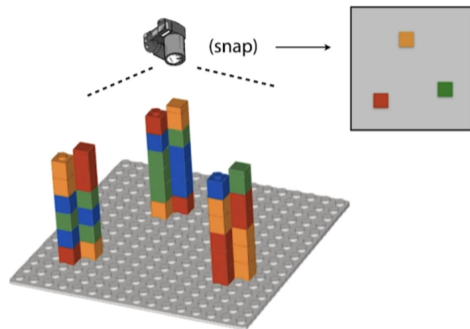
Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

Process (Basic Idea): when bases with terminators bind, no further base can be added.

terminators are engineered to glow a particular color (*A, C, G, T*)

→ take photo

after taking photo, terminators are removed and process is repeated.



Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences

- ▶ Several methods exist, one is the "Illumina sequencing process":

Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

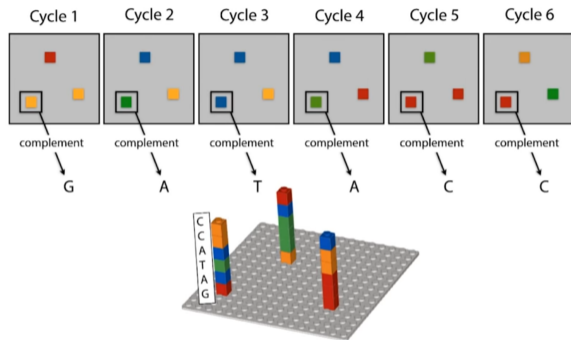
Process (Basic Idea): when bases with terminators bind, no further base can be added.

terminators are engineered to glow a particular color (A, C, G, T)

→ take photo

after taking photo, terminators are removed and process is repeated.

Output: the read of the **multiple** input DNAs (photos of each cycle)



Next-generation sequencing (NGS)

- ▶ used to read **multiple** "small (500bp)" DNA sequences
- ▶ Several methods exists, one is the "Illumina sequencing process":

Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

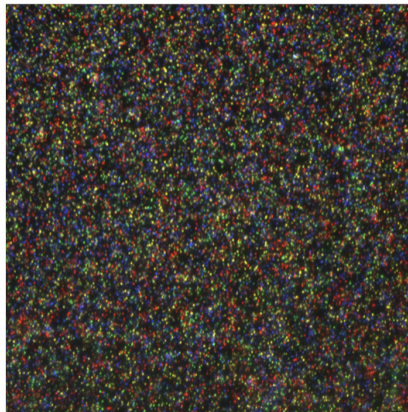
Process (Basic Idea): when bases with terminators bind, no further base can be added.

terminators are engineered to glow a particular color (*A, C, G, T*)

→ take photo

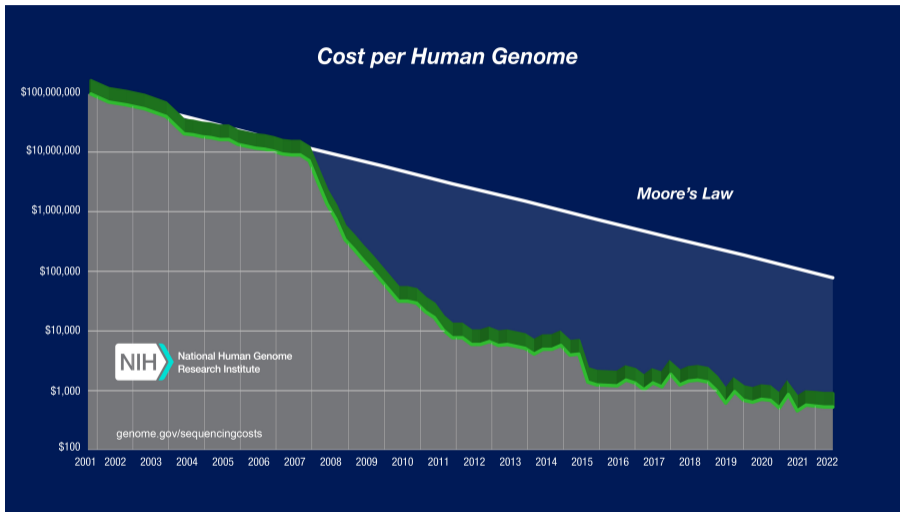
after taking photo, terminators are removed and process is repeated.

Output: the read of the **multiple** input DNAs (photos of each cycle)



Key feature:

massively parallel, photograph captures all templates simultaneously (billions of DNA templates on a slide)



- ▶ Copying DNA:
 - ▶ Polymerase chain reaction (PCR)
- ▶ Sequencing DNA:
 - ▶ Sanger Sequencing [AKA 1st generation sequencing]
 - ▶ Next/2nd-generation sequencing (NGS) [AKA Massive parallel sequencing]
 - ▶ 3rd-generation [AKA long-read sequencing]
(currently under active development*, can read more than 10000 bp)

To recall, human DNA 3.2×10^9 bp, Carsonella ruddii DNA 159 662bp

Observation: Whole genomes cannot be read at once.

*Marx, V. Method of the year: long-read sequencing. Nat Methods 20, 6–11 (2023).

- ▶ Copying DNA:
 - ▶ Polymerase chain reaction (PCR)
- ▶ Sequencing DNA:
 - ▶ Sanger Sequencing [AKA 1st generation sequencing]
 - ▶ Next/2nd-generation sequencing (NGS) [AKA Massive parallel sequencing]
 - ▶ 3rd-generation [AKA long-read sequencing]
(currently under active development*, can read more than 10000 bp)

To recall, human DNA 3.2×10^9 bp, Carsonella ruddii DNA 159 662bp

Observation: Whole genomes cannot be read at once.

*Marx, V. Method of the year: long-read sequencing. Nat Methods 20, 6–11 (2023).

unknown DNA
????????????????????????????????

Observation:

Sequencers cannot read whole genomes at once.

unknown DNA
????????????????????????????????

Observation:

Sequencers cannot read whole genomes at once.

Idea_1: randomly break-up long DNA into multiple pieces
(e.g. with ultrasound)
and sequence them



unknown DNA
????????????????????????????

Observation:

Sequencers cannot read whole genomes at once.

Idea_1: randomly break-up long DNA into multiple pieces
(e.g. with ultrasound)
and sequence them



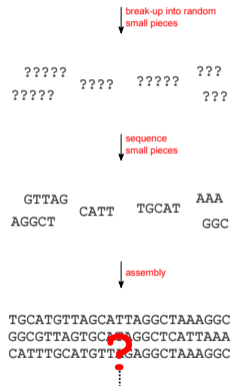
unknown DNA
????????????????????????????

Observation:

Sequencers cannot read whole genomes at once.

Idea_1: randomly break-up long DNA into multiple pieces
(e.g. with ultrasound)
and sequence them

However: if we just use a single DNA strand, well ..



Observation:

Sequencers cannot read whole genomes at once.

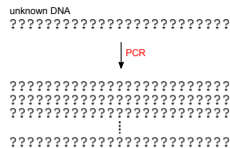
Idea_1: randomly break-up long DNA into multiple pieces

(e.g. with ultrasound)

and sequence them

However: if we just use a single DNA strand, well ..

Idea_2: Produce multiple copies of DNA first and then apply **Idea_1**



Observation:

Sequencers cannot read whole genomes at once.

Idea_1: randomly break-up long DNA into multiple pieces

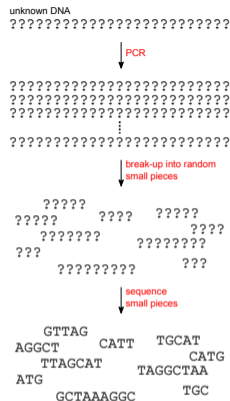
(e.g. with ultrasound)

and sequence them

However: if we just use a single DNA strand, well ..

Idea_2: Produce multiple copies of DNA first and then apply **Idea_1**

⇒ results in overlapping reads



Observation:

Sequencers cannot read whole genomes at once.

Idea_1: randomly break-up long DNA into multiple pieces

(e.g. with ultrasound)

and sequence them

However: if we just use a single DNA strand, well ..

Idea_2: Produce multiple copies of DNA first and then apply **Idea_1**

⇒ results in overlapping reads

⇒ assembly (here smart computational methods are needed!)



For a given set $\zeta = \{S_1, \dots, S_N\}$ of strings (=reads of fragments of DNA D), a superstring is a string S that contains all S_i as substrings.

Trivially, we could concatenate all strings in ζ to get superstring S . However, having say $\sim 10^6$ copies of DNA D fragmented and sequenced, we get then a string S of length $|S| \sim |D| \times 10^6$
 \implies far away from D .

In the assembly problem, we want to find a superstring that "best represents" D .

There are several ways on how to define "best represents" !!

We start with considering following problem:

Shortest Common Superstring Problem (SCS):

For a given $\zeta = \{S_1, \dots, S_N\}$ find a superstring S of shortest length.

SCS is NP-hard. So we focus ways to approximate solutions

\implies overlap graphs and Greedy_SCS + DeBruijn-graphs and Eulerian Paths
(board)

For a given set $\zeta = \{S_1, \dots, S_N\}$ of strings (=reads of fragments of DNA D), a superstring is a string S that contains all S_i as substrings.

Trivially, we could concatenate all strings in ζ to get superstring S . However, having say $\sim 10^6$ copies of DNA D fragmented and sequenced, we get then a string S of length $|S| \sim |D| \times 10^6$
 \implies far away from D .

In the assembly problem, we want to find a superstring that "best represents" D .

There are several ways on how to define "best represents" !!

We start with considering following problem:

Shortest Common Superstring Problem (SCS):

For a given $\zeta = \{S_1, \dots, S_N\}$ find a superstring S of shortest length.

SCS is NP-hard. So we focus ways to approximate solutions

\implies overlap graphs and Greedy_SCS + DeBruijn-graphs and Eulerian Paths
(board)

For a given set $\zeta = \{S_1, \dots, S_N\}$ of strings (=reads of fragments of DNA D), a superstring is a string S that contains all S_i as substrings.

Trivially, we could concatenate all strings in ζ to get superstring S . However, having say $\sim 10^6$ copies of DNA D fragmented and sequenced, we get then a string S of length $|S| \sim |D| \times 10^6 \implies$ far away from D .

In the assembly problem, we want to find a superstring that "best represents" D .

There are several ways on how to define "best represents" !!

We start with considering following problem:

Shortest Common Superstring Problem (SCS):

For a given $\zeta = \{S_1, \dots, S_N\}$ find a superstring S of shortest length.

SCS is NP-hard. So we focus ways to approximate solutions

\implies overlap graphs and Greedy_SCS + DeBruijn-graphs and Eulerian Paths
(board)

For a given set $\zeta = \{S_1, \dots, S_N\}$ of strings (=reads of fragments of DNA D), a superstring is a string S that contains all S_i as substrings.

Trivially, we could concatenate all strings in ζ to get superstring S . However, having say $\sim 10^6$ copies of DNA D fragmented and sequenced, we get then a string S of length $|S| \sim |D| \times 10^6 \implies$ far away from D .

In the assembly problem, we want to find a superstring that "best represents" D .

There are several ways on how to define "best represents" !!

We start with considering following problem:

Shortest Common Superstring Problem (SCS):

For a given $\zeta = \{S_1, \dots, S_N\}$ find a superstring S of shortest length.

SCS is NP-hard. So we focus ways to approximate solutions

\implies overlap graphs and Greedy_SCS + DeBruijn-graphs and Eulerian Paths
(board)

For a given set $\zeta = \{S_1, \dots, S_N\}$ of strings (=reads of fragments of DNA D), a superstring is a string S that contains all S_i as substrings.

Trivially, we could concatenate all strings in ζ to get superstring S . However, having say $\sim 10^6$ copies of DNA D fragmented and sequenced, we get then a string S of length $|S| \sim |D| \times 10^6 \implies$ far away from D .

In the assembly problem, we want to find a superstring that "best represents" D .

There are several ways on how to define "best represents" !!

We start with considering following problem:

Shortest Common Superstring Problem (SCS):

For a given $\zeta = \{S_1, \dots, S_N\}$ find a superstring S of shortest length.

SCS is NP-hard. So we focus ways to approximate solutions

\implies overlap graphs and Greedy_SCS + DeBruijn-graphs and Eulerian Paths
(board)

Genomes often consist of repeated regions!

Example: Here, ζ = set of all substrings of size 6 and $|\zeta| = 16$ for all examples.

Greedy SCS on 6-mers of **a_long_long_time**

```
ng_lon _long_ a_long long_l ong_ti ong_lo long_t g_long g_time ng_tim
ng_time ng_lon _long_ a_long long_l ong_ti ong_lo long_t g_long
ng_time g_long_ ng_lon a_long long_l ong_ti ong_lo long_t
ng_time long_ti g_long_ ng_lon a_long long_l ong_lo
ng_time ong_lon long_ti g_long_ a_long long_l
ong_lon long_time g_long_ a_long long_l
long_lon long_time g_long_ a_long
long_lon g_long_time a_long
long_long_time a_long
a_long_long_time
```

Genomes often consist of repeated regions!

Example: Here, ζ = set of all substrings of size 6 and $|\zeta| = 16$ for all examples.

Greedy SCS on 6-mers of **a_long_long_time**

```
ng_lon _long_ a_long long_l ong_ti ong_lo long_t g_long g_time ng_tim
ng_time ng_lon _long_ a_long long_l ong_ti ong_lo long_t g_long
ng_time g_long_ ng_lon a_long long_l ong_ti ong_lo long_t
ng_time long_ti g_long_ ng_lon a_long long_l ong_lo
ng_time ong_lon long_ti g_long_ a_long long_l
ong_lon long_time g_long_ a_long long_l
long_lon long_time g_long_ a_long
long_lon g_long_time a_long
long_long_time a_long
a_long_long_time
```

The final superstring is shorter than the original "genome"

Genomes often consist of repeated regions!

Example: Here, ζ = set of all substrings of size 6 and $|\zeta| = 16$ for all examples.

```
a_long_long_time    a_long_long_long_time
a_long long_t       a_long long_l ng_tim
_long_ong_ti        _long_ong_lo g_time
_long_l ng_tim      long_l ng_lon
long_l g_time       ong_lo g_long
ong_lo              ng_lon _long_
ong_lo              g_long long_t
ng_lon              _long_ong_ti
ng_lon
g_long
g_long
_long_
_long_

a_long_long_long_long_long_time
a_long long_l g_long ng_tim
_long_ ng_lon long_l g_time
ong_lo _long_ ng_lon
g_long ong_lo _long_
long_t
ong_ti
```

Genomes often consist of repeated regions!

Example: Here, ζ = set of all substrings of size 6 and $|\zeta| = 16$ for all examples.

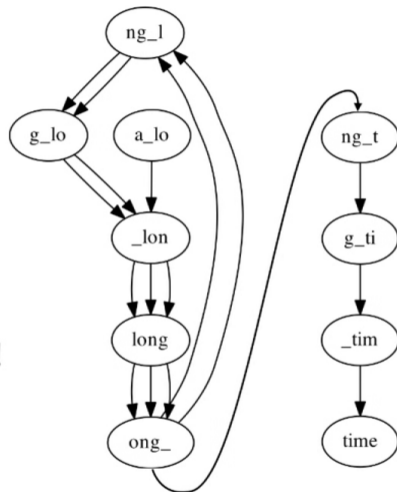
```
a_long_long_time    a_long_long_long_time
a_long long_t       a_long long_l ng_tim
_long_ong_ti        _long_ong_lo g_time
_long_l ng_tim      long_l ng_lon
long_l g_time       ong_lo g_long
ong_lo              ng_lon _long_
ong_lo              g_long long_t
ng_lon              _long_ong_ti
ng_lon
g_long
g_long
_long_
_long_

a_long_long_long_long_long_time
a_long long_l g_long ng_tim
_long_ ng_lon long_l g_time
ong_lo _long_ ng_lon
g_long ong_lo _long_
long_t
ong_ti
```

To work with such problems one may employ: DeBruijn-graphs and Eulerian Paths. (board)

De Bruijn graph ($k=5$) for:
a_long_long_long_time

Eulerian walk gives original genome!

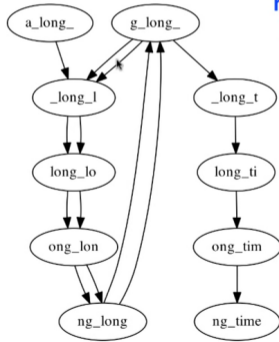


The Assembly Problem

$k=8$ Genome: a_long_long_long_time

Reads: a_long_long_long, ng_long_l, g_long_time

k-mers: a_long_l ng_long_l g_long_t
 _long_lo g_long_l _long_ti
 _long_lon _long_lo _long_tim
 ong_long ong_long ong_time
 ng_long ng_long_l
 g_long_l
 _long_lo
 _long_lon
 ong_long



No Eulerian Walk!

Overlap graphs and DeBruijn graphs can be used to represent "relationships" between substrings.

The provided algorithms can, in general, not solve the assembly problem in an "optimal way" but serve as useful heuristics.

There are more sophisticated methods out there that are often based on these type of algorithms that of often based on the latter ideas.

*Medvedev & Pop *What do Eulerian and Hamiltonian cycles have to do with genome assembly?* PLoS Comput Biol. 2021

Classical problems in practice:

- ▶ sequencing errors
- ▶ overlapping regions of fragments that are located on "far away" positions on DNA
- ▶ incomplete data (DNA not covered by resulting sequenced fragments)
- ▶ orientation of reads usually unknown
- ▶ repeats