Computational Biology

DNA Sequencing

Department of Mathematics
Stockholm University

In a nutshell: Breakthrough Technologies

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

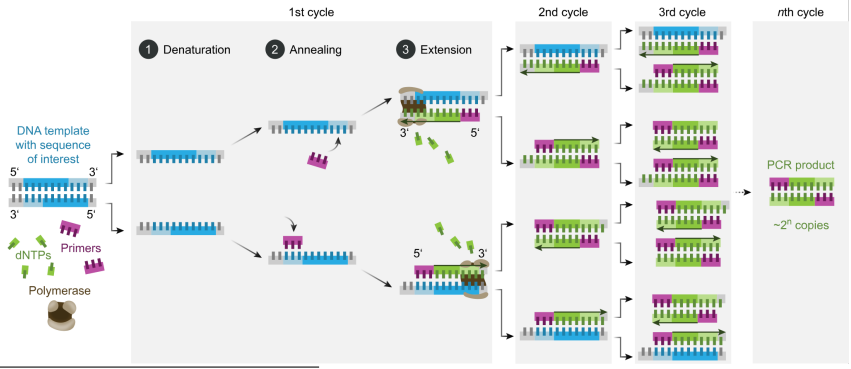


fig taken from https://en.wikipedia.org/wiki/Polymerase_chain_reaction

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

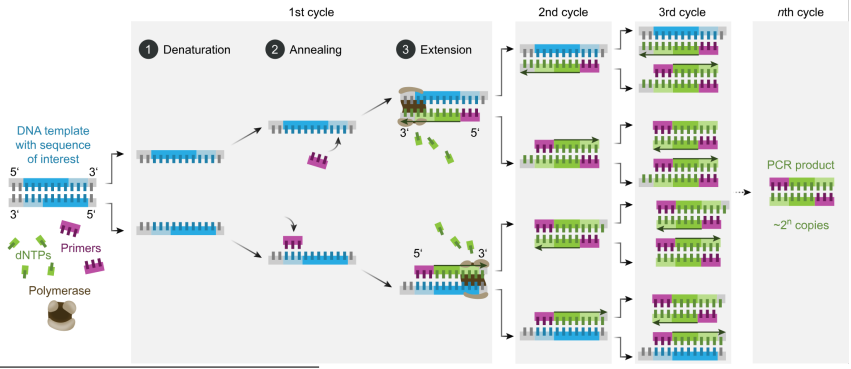


fig taken from https://en.wikipedia.org/wiki/Polymerase_chain_reaction

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

fig taken from https://www.frontiersin.org/articles/10.3389/fmicb.2021.766364/full

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

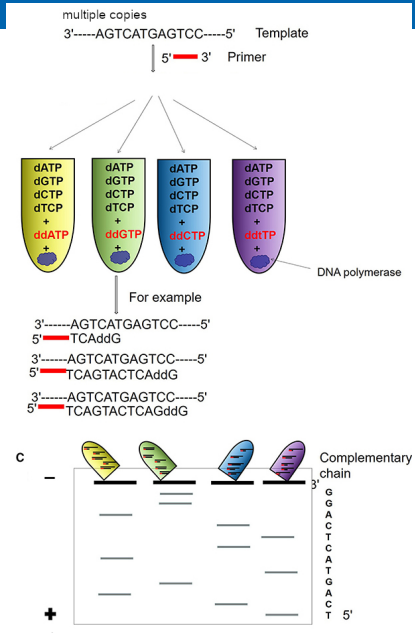


fig taken from https://www.frontiersin.org/articles/10.3389/fmicb.2021.766364/full

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

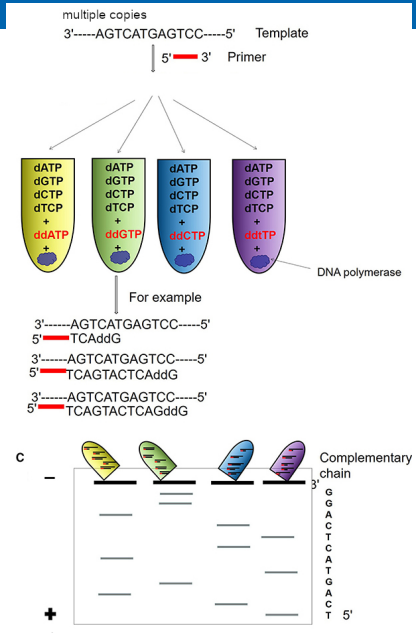


fig taken from https://www.frontiersin.org/articles/10.3389/fmicb.2021.766364/full

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

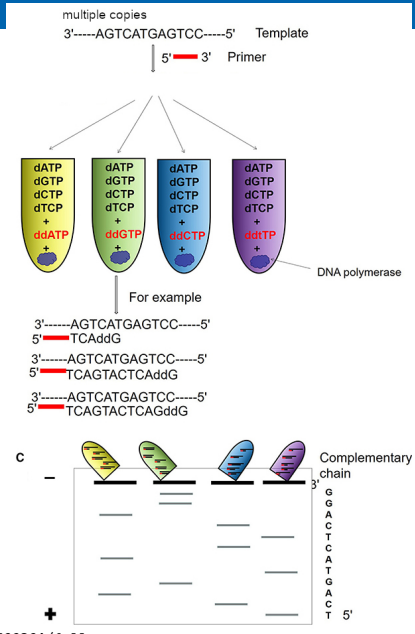


fig taken from https://www.frontiersin.org/articles/10.3389/fmicb.2021.766364/full

Next-generation sequencing (NGS)

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

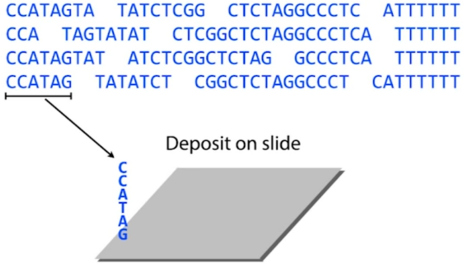
  Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

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

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

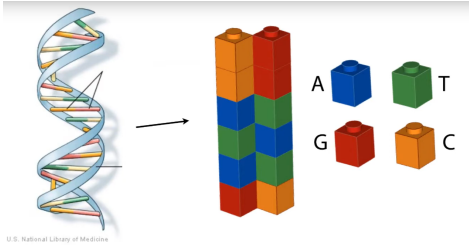


figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

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

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



Double stranded DNA (double helix)

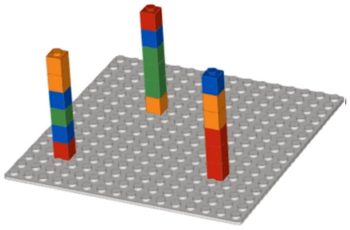
Double stranded DNA (lego version)

figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

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

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

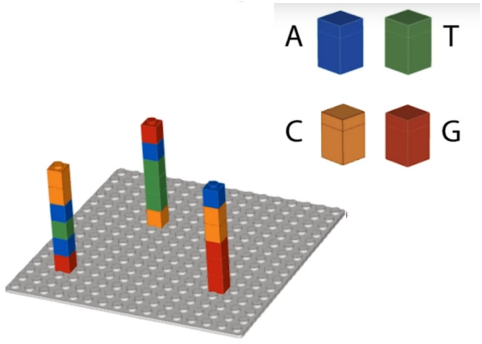


figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

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

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



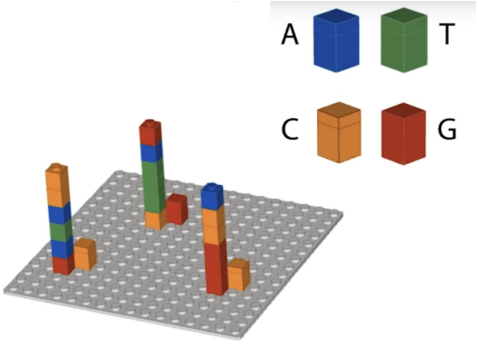
figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

- used to read **multiple** "small (500bp)" DNA sequences
- Several methods exits, 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.



figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

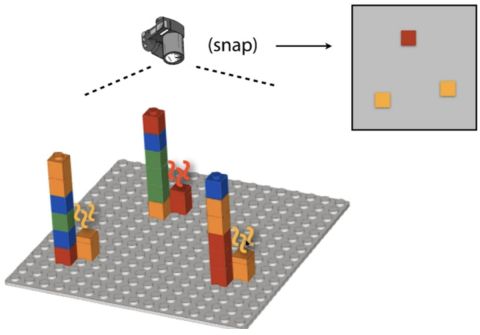
Next-generation sequencing (NGS)

- used to read **multiple** "small (500bp)" DNA sequences
- Several methods exits, 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)
  $\rightarrow$ take photo



figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

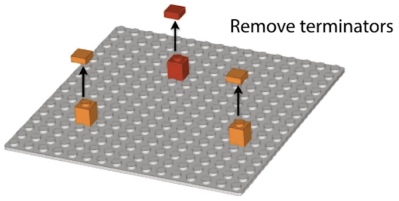
- used to read **multiple** "small (500bp)" DNA sequences
- Several methods exits, one is the "Illumina sequencing process":

  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)
  $\rightarrow$ take photo

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



figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

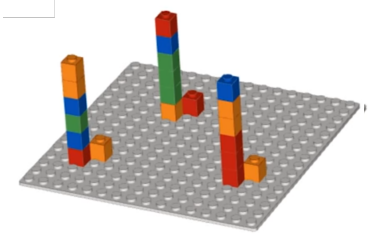
- used to read **multiple** "small (500bp)" DNA sequences
- Several methods exits, one is the "Illumina sequencing process":

  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)
  $\rightarrow$ take photo

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



figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

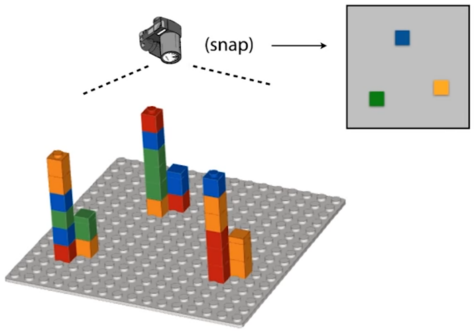
- used to read **multiple** "small (500bp)" DNA sequences
- Several methods exits, one is the "Illumina sequencing process":

  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)
  $\rightarrow$ take photo

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



figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

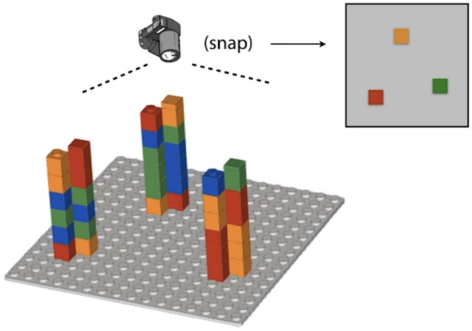
- used to read **multiple** "small (500bp)" DNA sequences
- Several methods exits, one is the "Illumina sequencing process":

  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)
  $\rightarrow$ take photo

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



figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

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

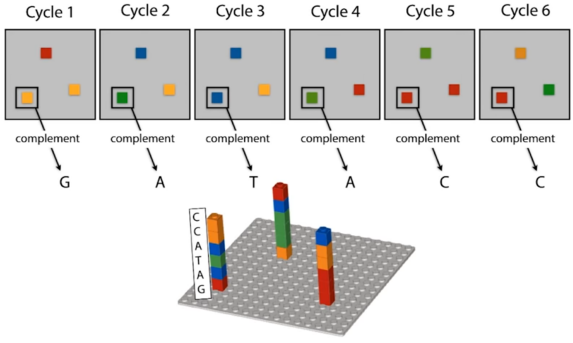
  Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

  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)
  $\rightarrow$ take photo

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

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



figs taken from Ben Langmead "ADS1: Sequencing by Synthesis" (youtube)

Next-generation sequencing (NGS)

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

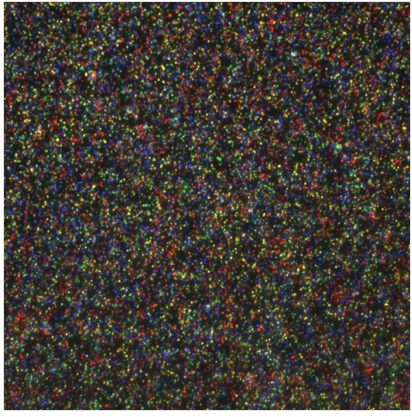
  Input: copies of **multiple** DNA (fragments) placed on a slide, bases, terminators, polymerase, ..

  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)
  $\rightarrow$ take photo

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

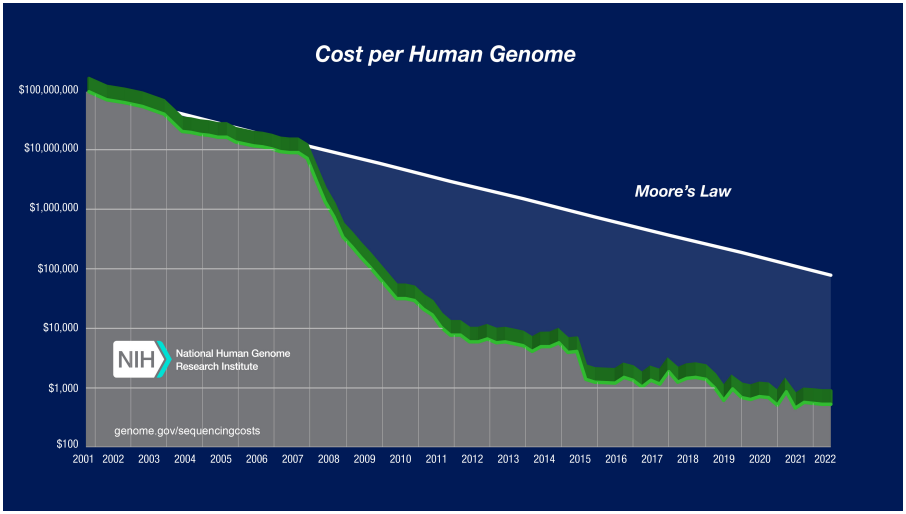
  Output: the read of the **multiple** input DNAs (photos of each cycle)



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

fig taken from `https://theory.labster.com/ngs-experiment/`

In a nutshell: Breakthrough Technologies



- 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, humanDNA 3.2×10^9bp, 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, humanDNA 3.2×10^9bp, 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).

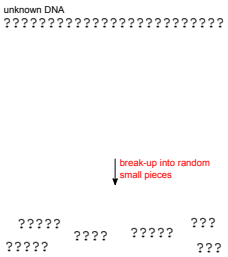
Shotgun Sequencing

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

Observation:

Sequencers cannot read whole genomes at once.

Shotgun Sequencing



Observation:

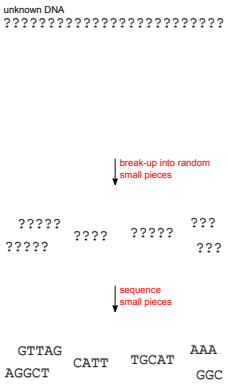
Sequencers cannot read whole genomes at once.

Idea_1: randomly break-up long DNA into multiple pieces

(e.g. with ultrasound)

and sequence them

Shotgun Sequencing



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

Shotgun Sequencing

Observation:

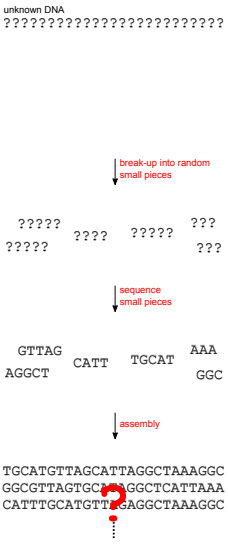
Sequencers cannot read whole genomes at once.

Idea_1: randomly break-up long DNA into multiple pieces

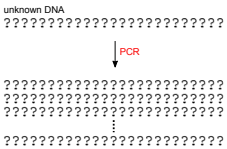
(e.g. with ultrasound)

and sequence them

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



Shotgun Sequencing



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

Shotgun Sequencing

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

$\Longrightarrow$ results in overlapping reads

Shotgun Sequencing

Observation:

Sequencers cannot read whole genomes at once.

Idea_1: randomly break-up long DNA into multiple pieces

(e.g. with ultrasound)

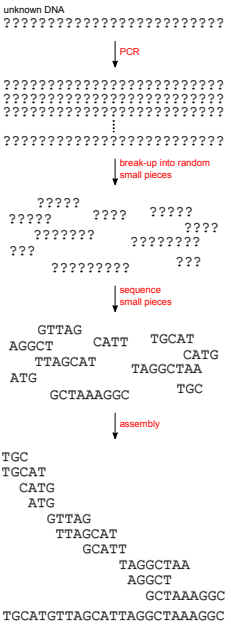
and sequence them

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

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

$\Longrightarrow$ results in overlapping reads

$\Longrightarrow$ assembly (here smart computational methods are needed!)



The Assembly Problem

For a given set $\zeta = \{S_1, \ldots S_N\}$ 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 $|S| \sim |D| \times 10^6$
$\Longrightarrow$ 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 $\zeta = \{S_1, \ldots S_N\}$ find a superstring S of shortest length.

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

$\Longrightarrow$ overlap graphs and `Greedy_SCS` + DeBruijn-graphs and Eulerian Paths
(board)

The Assembly Problem

For a given set $\zeta = \{S_1, \ldots S_N\}$ 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 $|S| \sim |D| \times 10^6$
$\Longrightarrow$ 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 $\zeta = \{S_1, \ldots S_N\}$ find a superstring S of shortest length.

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

$\Longrightarrow$ overlap graphs and `Greedy_SCS` + DeBruijn-graphs and Eulerian Paths
(board)

The Assembly Problem

For a given set $\zeta = \{S_1, \ldots S_N\}$ 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 $|S| \sim |D| \times 10^6$
$\Longrightarrow$ 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 $\zeta = \{S_1, \ldots S_N\}$ find a superstring S of shortest length.

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

$\Longrightarrow$ overlap graphs and `Greedy_SCS` + DeBruijn-graphs and Eulerian Paths

(board)

The Assembly Problem

For a given set $\zeta = \{S_1, \ldots S_N\}$ 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 $|S| \sim |D| \times 10^6$
$\Longrightarrow$ 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 $\zeta = \{S_1, \ldots S_N\}$ find a superstring S of shortest length.

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

$\Longrightarrow$ overlap graphs and `Greedy_SCS` + DeBruijn-graphs and Eulerian Paths
(board)

The Assembly Problem

For a given set $\zeta = \{S_1, \ldots S_N\}$ 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 $|S| \sim |D| \times 10^6$
$\Longrightarrow$ 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 $\zeta = \{S_1, \ldots S_N\}$ find a superstring S of shortest length.

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

$\Longrightarrow$ overlap graphs and `Greedy_SCS` + DeBruijn-graphs and Eulerian Paths

(board)

The Assembly Problem

Genomes often consist of repeated regions!

Example:

Greedy SCS on 6-mers of **a_long_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 Assembly Problem

Genomes often consist of repeated regions!

Example:

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

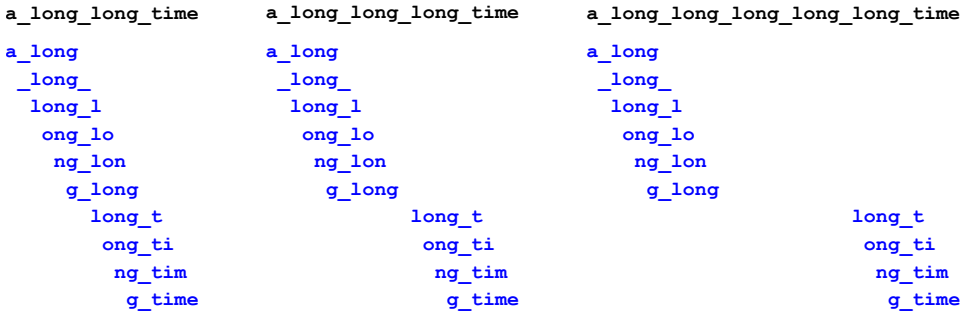
```
ng_lon _long_ a_long long_l ong_ti ong_lo long_t g_long g_time ng_tim
ng_time ng_lon _long_ a_long long_l ong_ti ong_lo long_t g_long
ng_time g_long_ ng_lon a_long long_l ong_ti ong_lo long_t
ng_time long_ti g_long_ ng_lon a_long long_l ong_lo
ng_time ong_lon long_ti g_long_ a_long long_l
ong_lon 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"

The Assembly Problem

Genomes often consist of repeated regions!

Example: For all three examples, choose ζ = set of all substrings of size 6 (identical for all!)



The Assembly Problem

Genomes often consist of repeated regions!

Example: For all three examples, choose ζ = set of all substrings of size 6 (identical for all!)

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

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

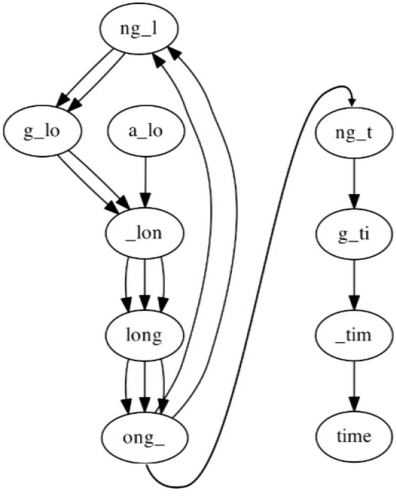
```
a_long_long_long_long_long_time
a_long
 _long_
  long_l
   ong_lo
    ng_lon
     g_long
                          long_t
                           ong_ti
                            ng_tim
                             g_time
```

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!

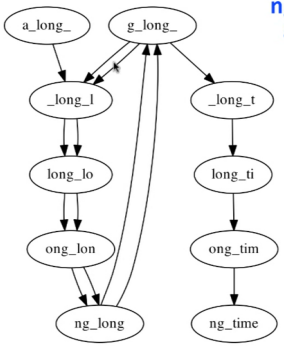


The Assembly Problem

$k = 8$ Genome: a_long_long_long_time

Reads: a_long_long_long, ng_long_l, g_long_time

k-mers: a_long_l _long_lo long_lon ong_long ng_long_ g_long_l _long_l long_lo long_lon ong_long g_long_t _long_ti long_tim ong_time



No Eulerian Walk!

The Assembly Problem

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