Population Genetics using Trees

Peter Beerli
Genome Sciences
University of Washington
Seattle WA

Outline

1. **Introduction to the basic coalescent**
   - Population models
   - The coalescent
   - Likelihood estimation of parameters of interest
   - Why do we need Markov chain Monte Carlo

2. **Extensions and examples**

Population genetics can help us to find answers

- the PCR revolution allows us to generate lots of data from many individuals and many loci
- We are still interested in questions like
  - Where are we or other species coming from?
  - How big are populations?
  - Are these populations species?
  - What is the recombination rate in species x?

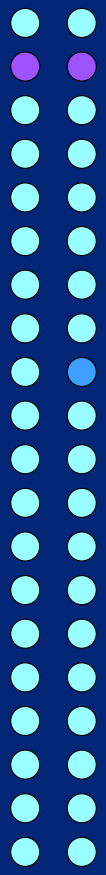
Population genetics in the age of genomics

- Why do we need theoretical population genetics when we can have the complete sequences of our favorite organism?

Basics: Wright-Fisher population model

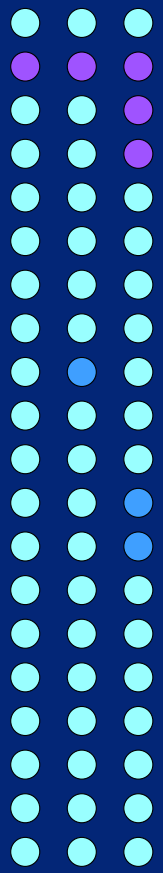
All individuals release many gametes and new individuals for the next generation are formed randomly from these.

Basics: Wright-Fisher population model



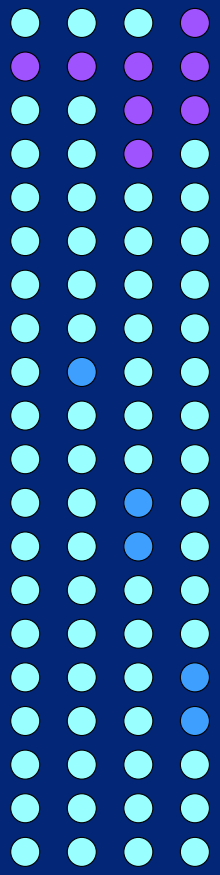
All individuals release many gametes and new individuals for the next generation are formed randomly from these.

Basics: Wright-Fisher population model



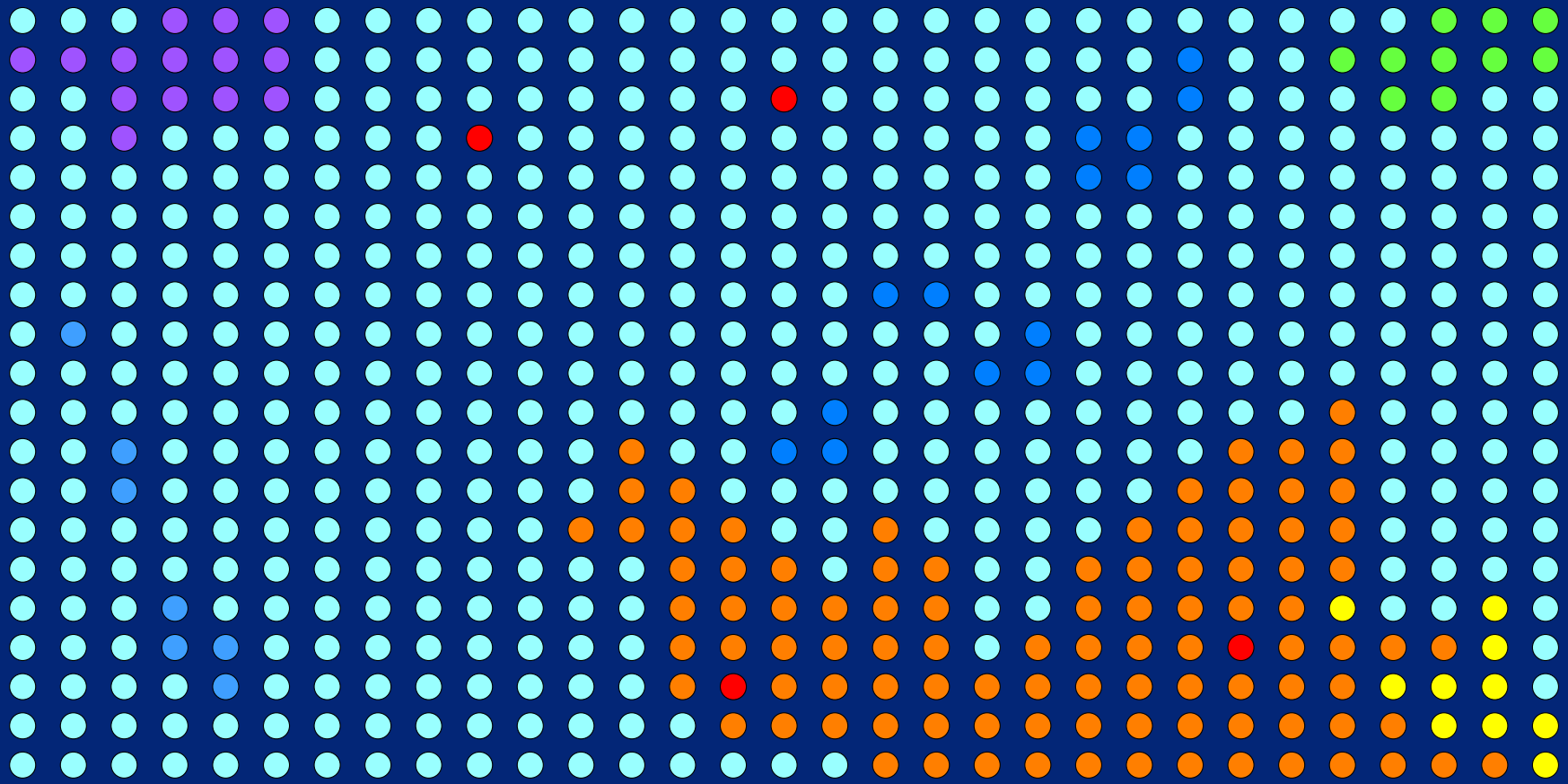
All individuals release many gametes and new individuals for the next generation are formed randomly from these.

Basics: Wright-Fisher population model



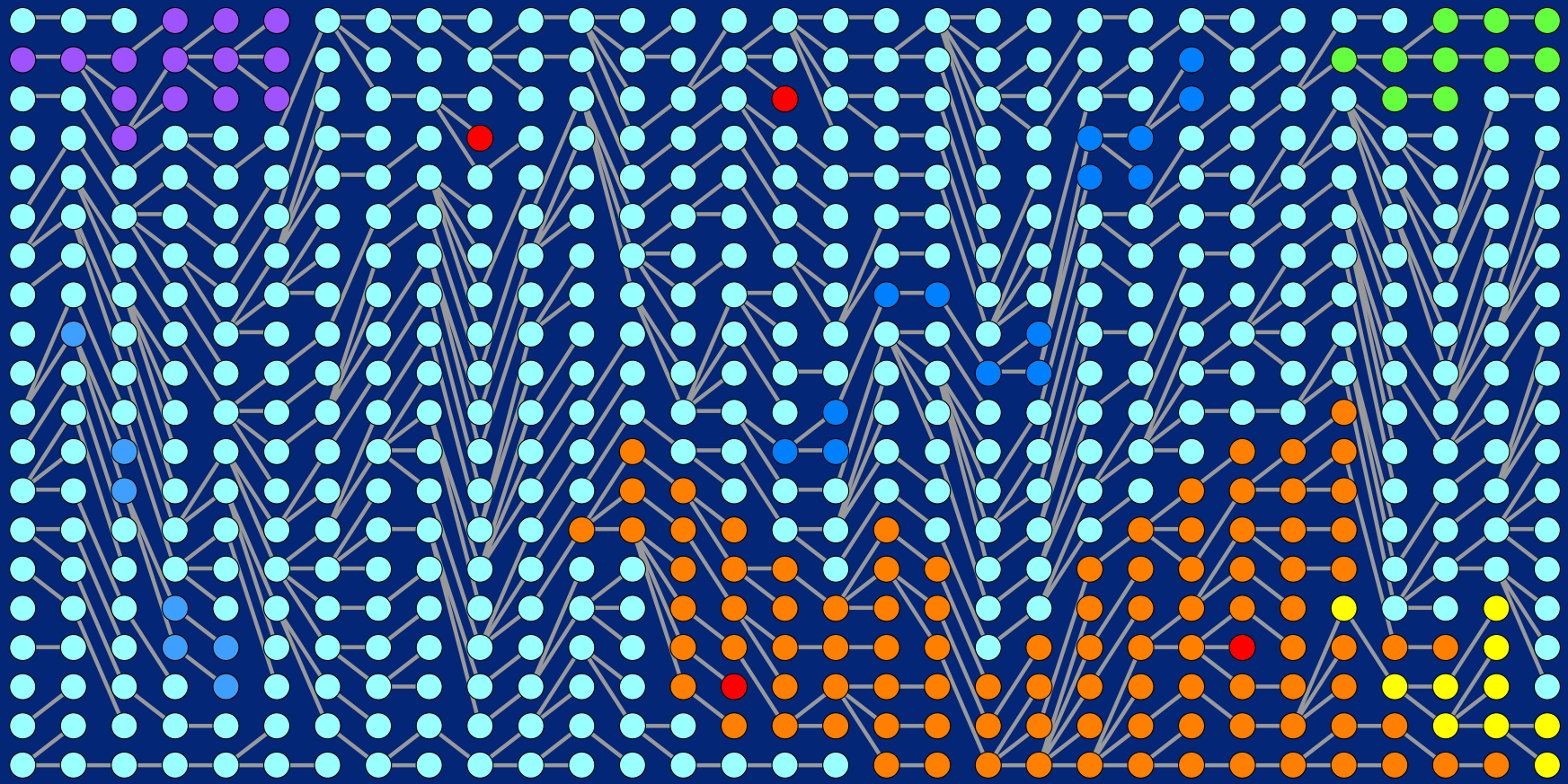
All individuals release many gametes and new individuals for the next generation are formed randomly from these.

Basics: Wright-Fisher population model



All individuals release many gametes and new individuals for the next generation are formed randomly from these.

Basics: Wright-Fisher population model

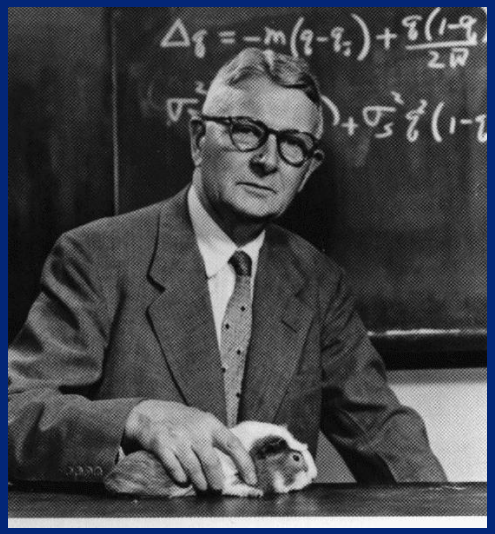


All individuals release many gametes and new individuals for the next generation are formed randomly from these.

Wright-Fisher population model

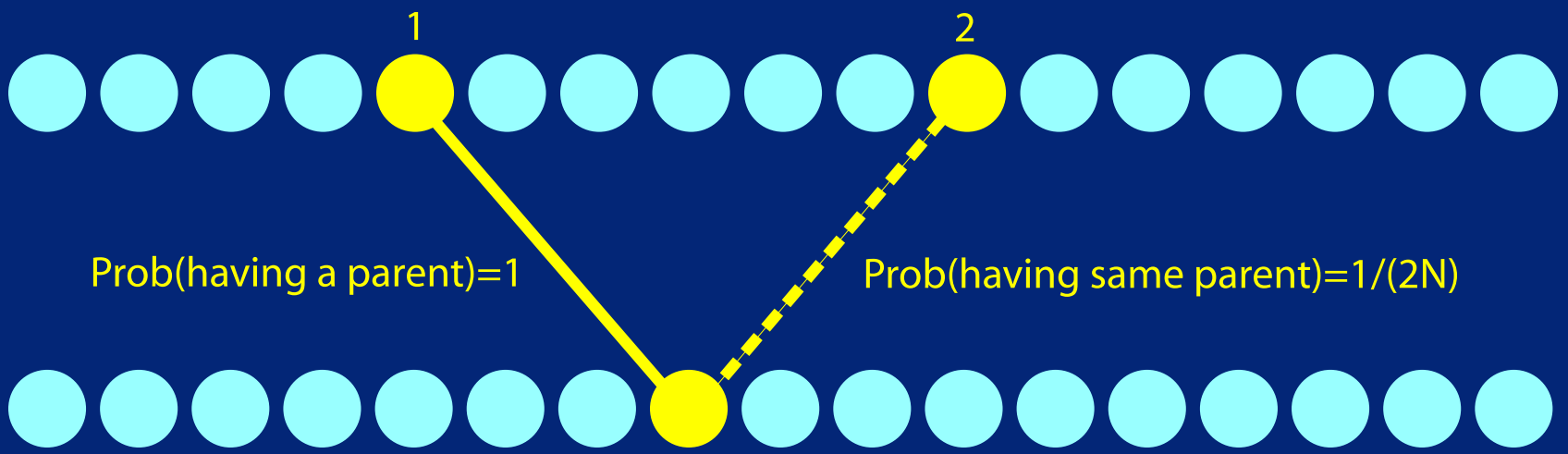
- Population size N is constant through time.
- Each individual gets replaced every generation.
- Next generation is drawn randomly from a large gamete pool.
- Only genetic drift is manipulating the allele frequencies.

The *Coalescent*

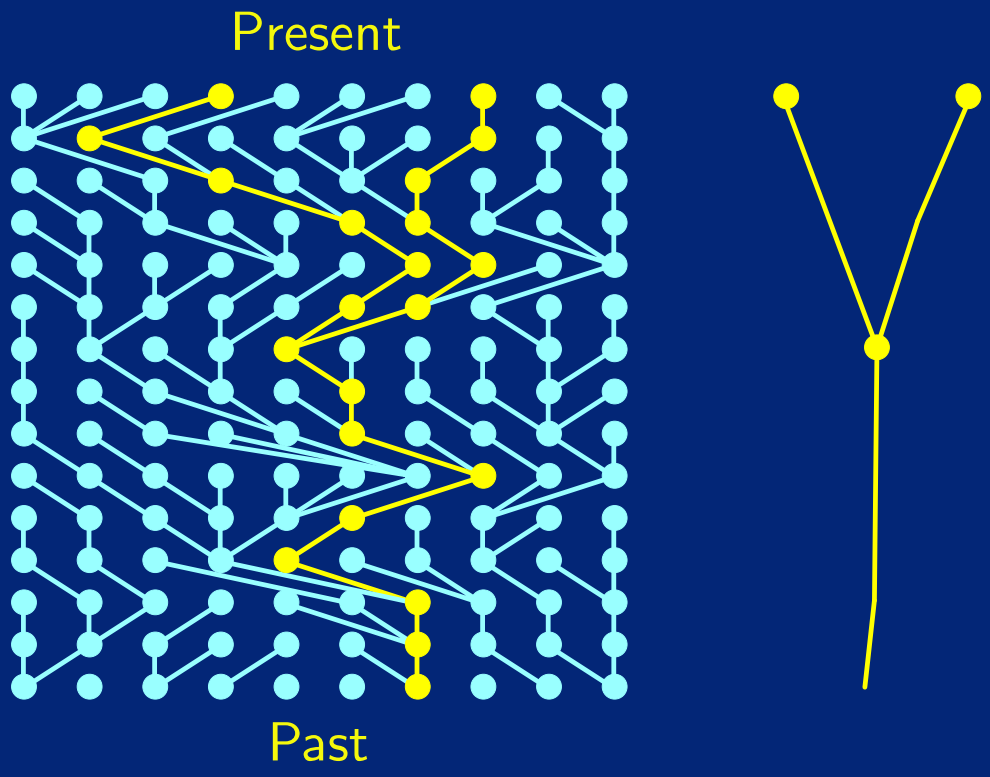


Sewall Wright showed that the probability that 2 gene copies come from the same gene copy in the preceding generation is

$$\text{Prob (two genes share a parent)} = \frac{1}{2N}$$



The *Coalescent*



In every generation, there is a chance of $1/2N$ to coalesce. Following the sampled lineages through generations backwards in time we realize that it follows a geometric distribution with

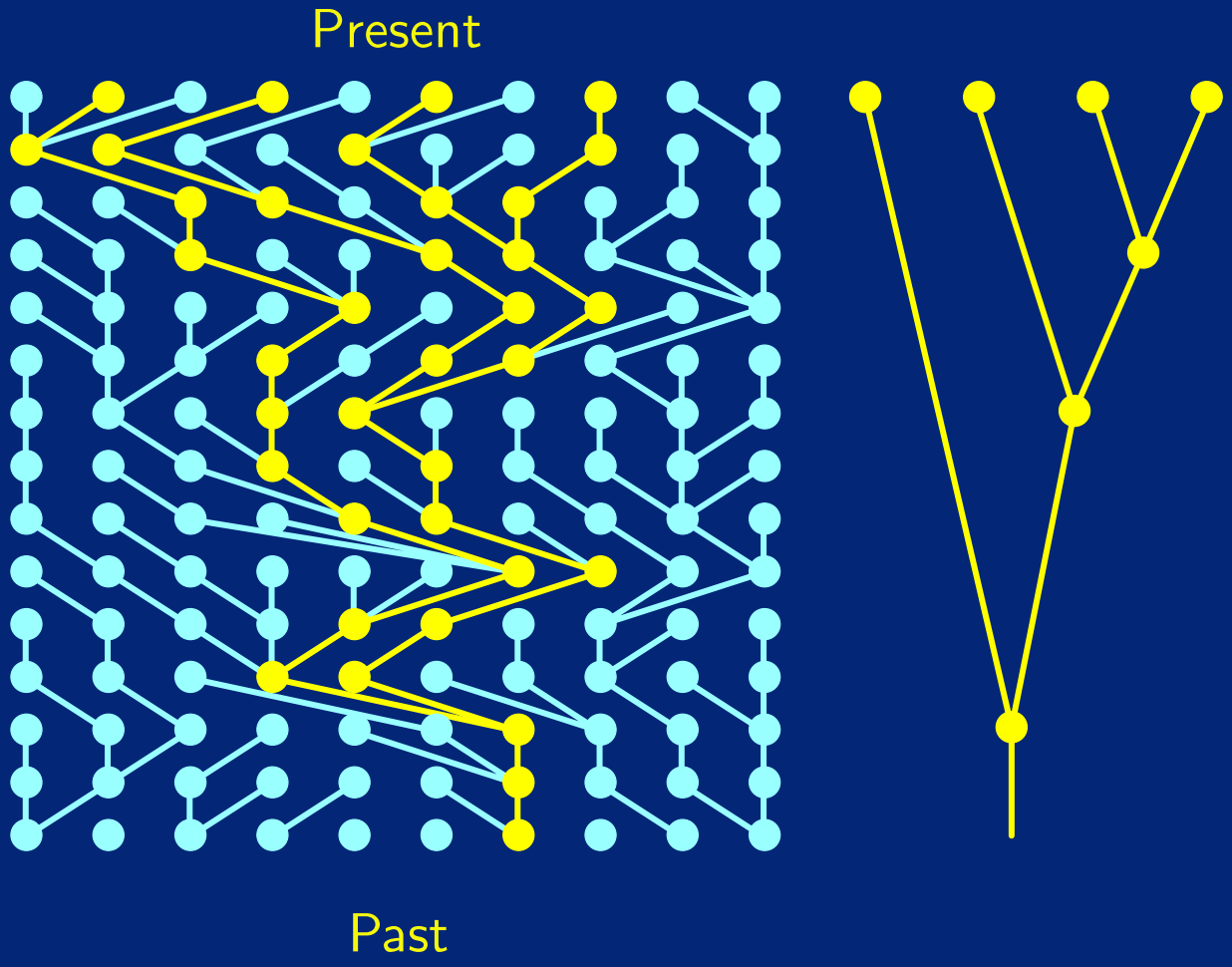
$\mathbb{E}(u) = 2N$ [the expectation of the time of coalescence u of **two** tips is $2N$]

The *Coalescent*

JFC Kingman generalized this for k gene copies.

$$\text{Prob } (k \text{ copies are reduced to } k-1 \text{ copies}) = \frac{k(k-1)}{4N}$$

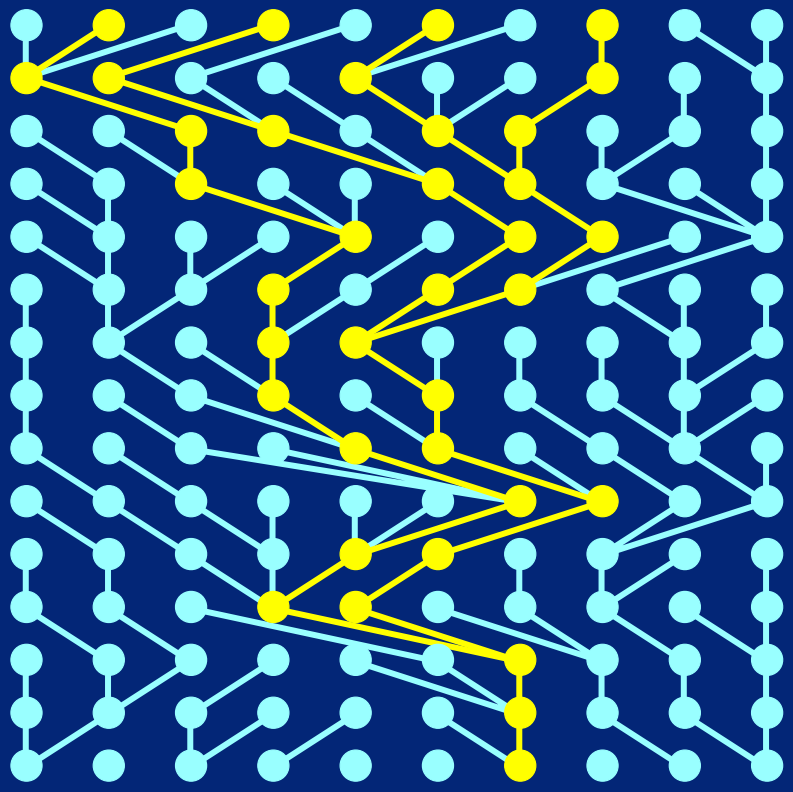
Kingman's n-coalescent

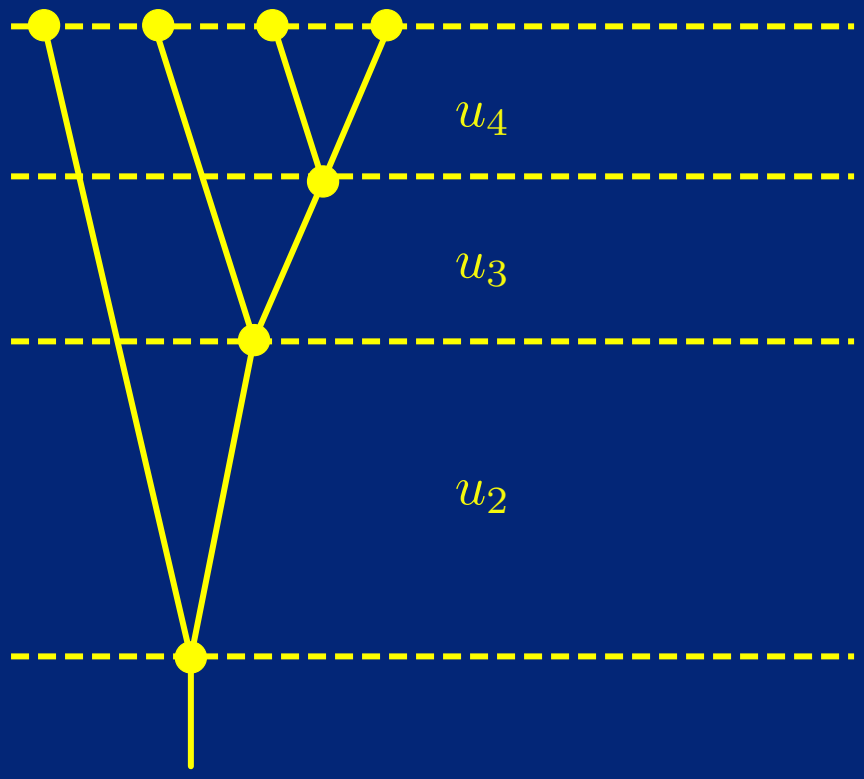


Kingman's n-coalescent

The expectation for the time interval u_k is

$$\mathbb{E}(u_k) = \frac{4N}{k(k-1)}$$





$$\mathrm{p(G|N)} = \prod_{\mathrm{i}} \exp(-\mathrm{u_i}\frac{\mathrm{k(k-1)}}{\mathrm{4N}})\frac{1}{\mathrm{2N}}$$

BUT, what's this good for??????????????????

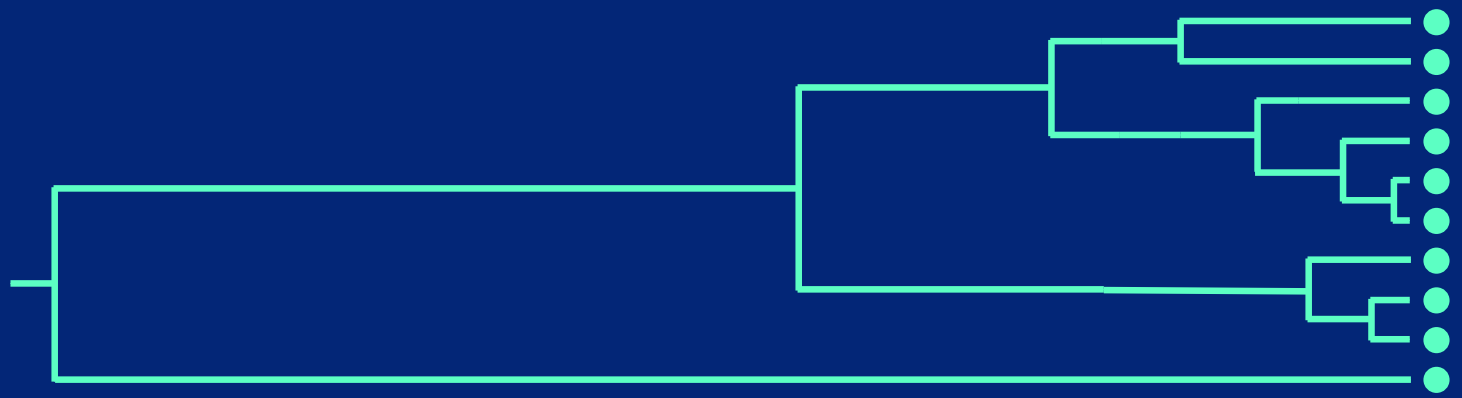
Naively we could estimate: 1.Time of the most recent common ancestor

For a given population size we can calculate the time of the most recent common ancestor [MRCA].

1. Get a TRUE genealogy (topology and branch lengths) from an infallible oracle.

2. Get the population size from the same oracle.

3. Calculate the time of the MRCA by summing over all time intervals.



1. Time of the most recent common ancestor [Shortcut]

1. Get the population size from another oracle

2. Use the expectation for your data type to get an estimate of the time of the MRCA

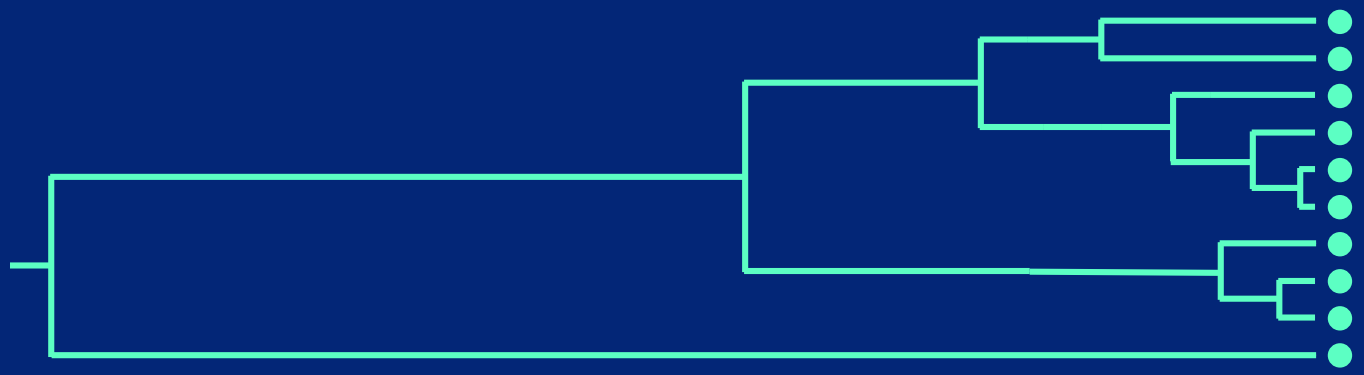
The expectation for the time of the MRCA is

$\mathbb{E}(u) = 4N$	for diploid organisms
$\mathbb{E}(u) = 2N$	for haploid organisms
$\mathbb{E}(u) = N$	for maternally transmitted mtDNA, paternally transmitted Y-chromosome [assumption: sex-ratio is 1:1]

2. Calculate the size of the population

1. We get THE genealogy from our oracle

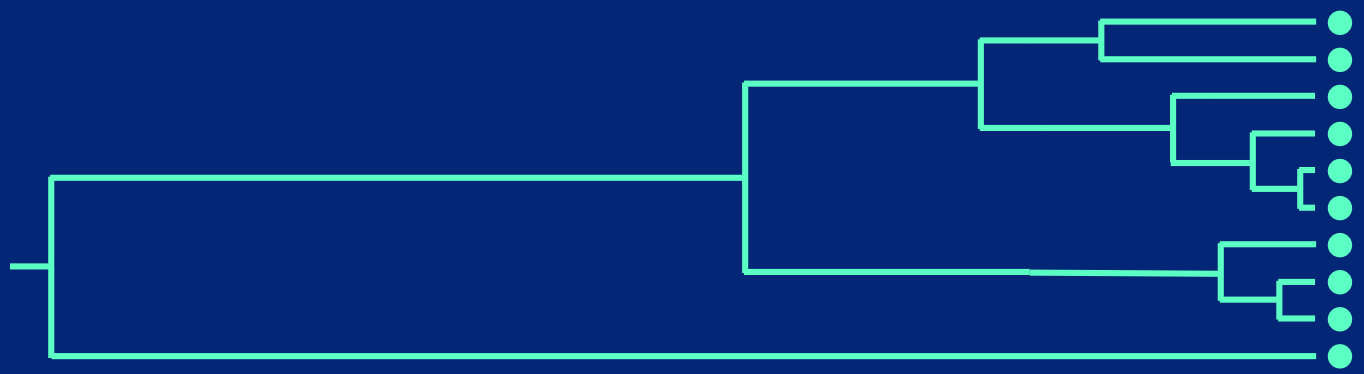
2. We know that we can calculate $\mathrm{p}(\text{Genealogy}|\mathrm{N})$



2. Calculate the size of the population

1. We get THE genealogy from our oracle

2. We remember the probability calculation

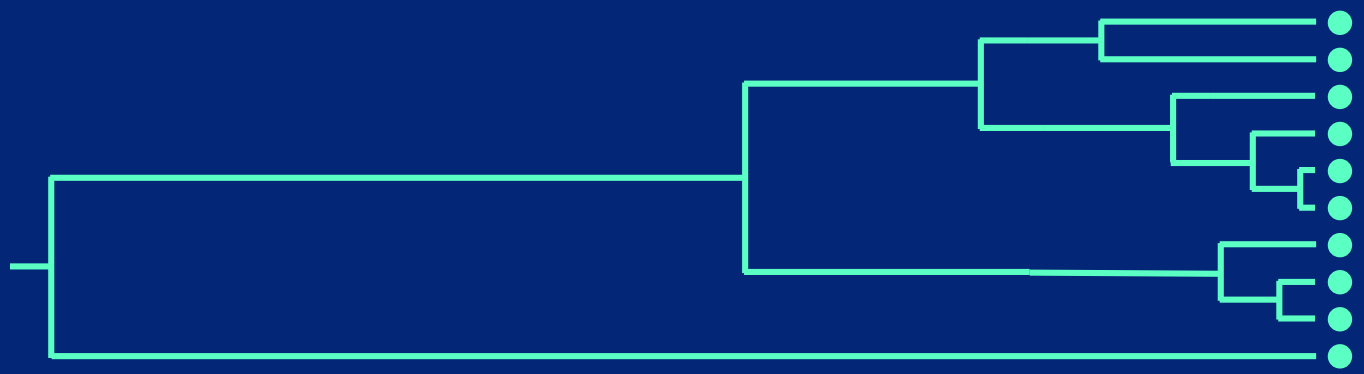


$$\mathrm{p(G|N)} = \mathrm{p(u_1|N,k)}\frac{1}{2\mathrm{N}} \times \mathrm{p(u_2|N,k-1)}\frac{1}{2\mathrm{N}} \times .....$$

2. Calculate the size of the population

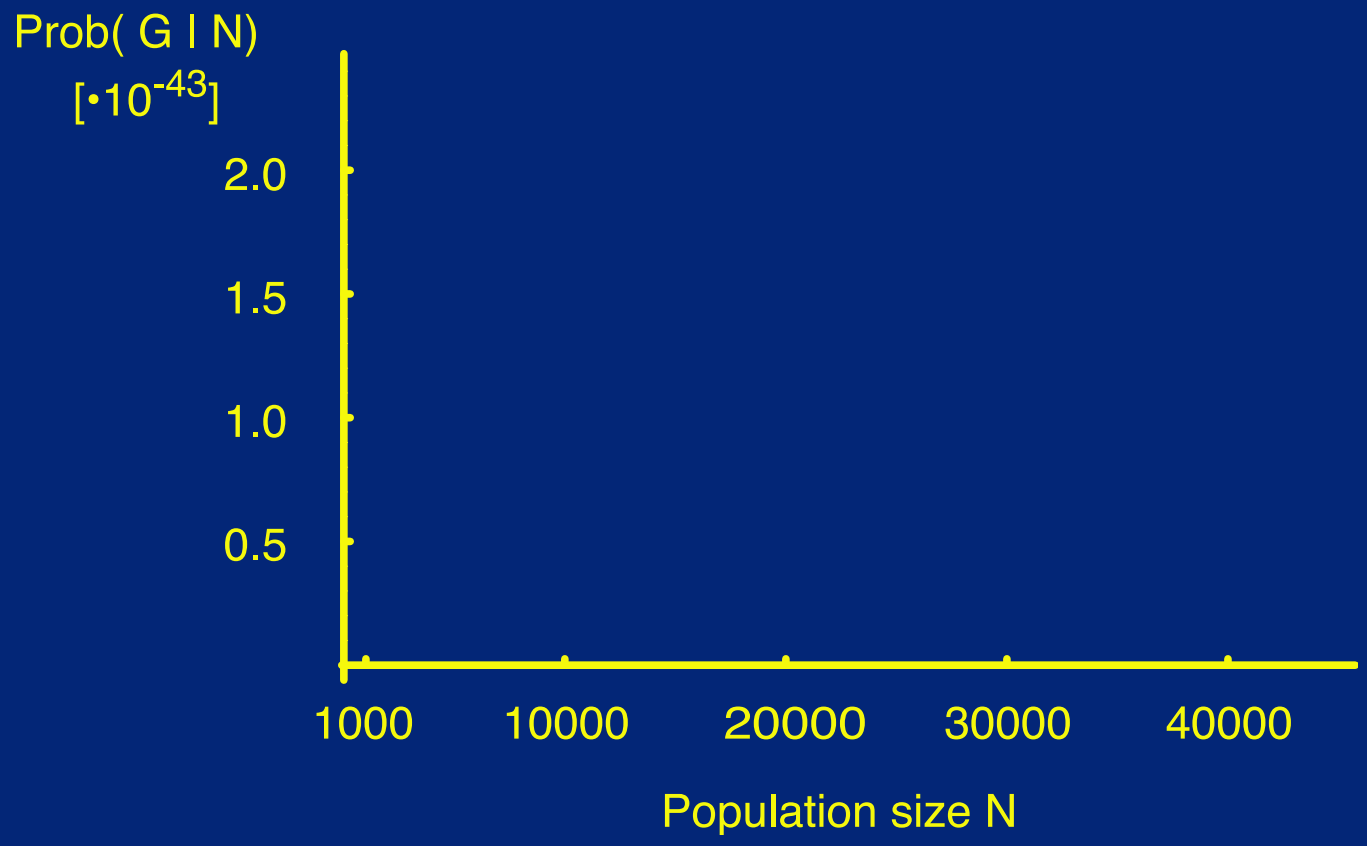
1. We get THE genealogy from our oracle

2. We remember the probability calculation

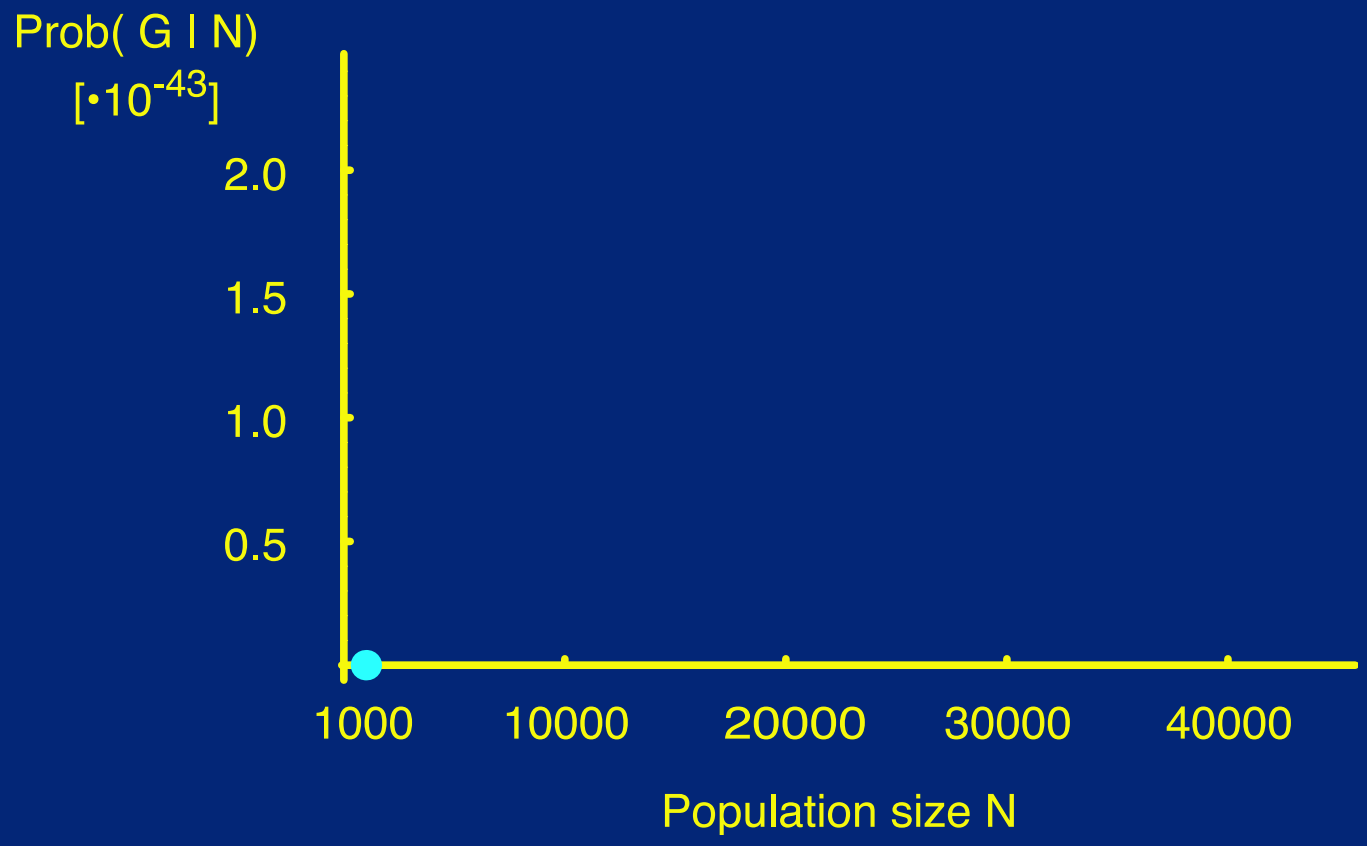


$$\mathrm{p}(\text{Genealogy}|\mathrm{N}) = \prod_{\mathrm{j}}^{\mathrm{T}} \mathrm{e}^{-\mathrm{u_j}\frac{\mathrm{k_j}(\mathrm{k_j}-1)}{4\mathrm{N}}} \frac{1}{2\mathrm{N}}$$

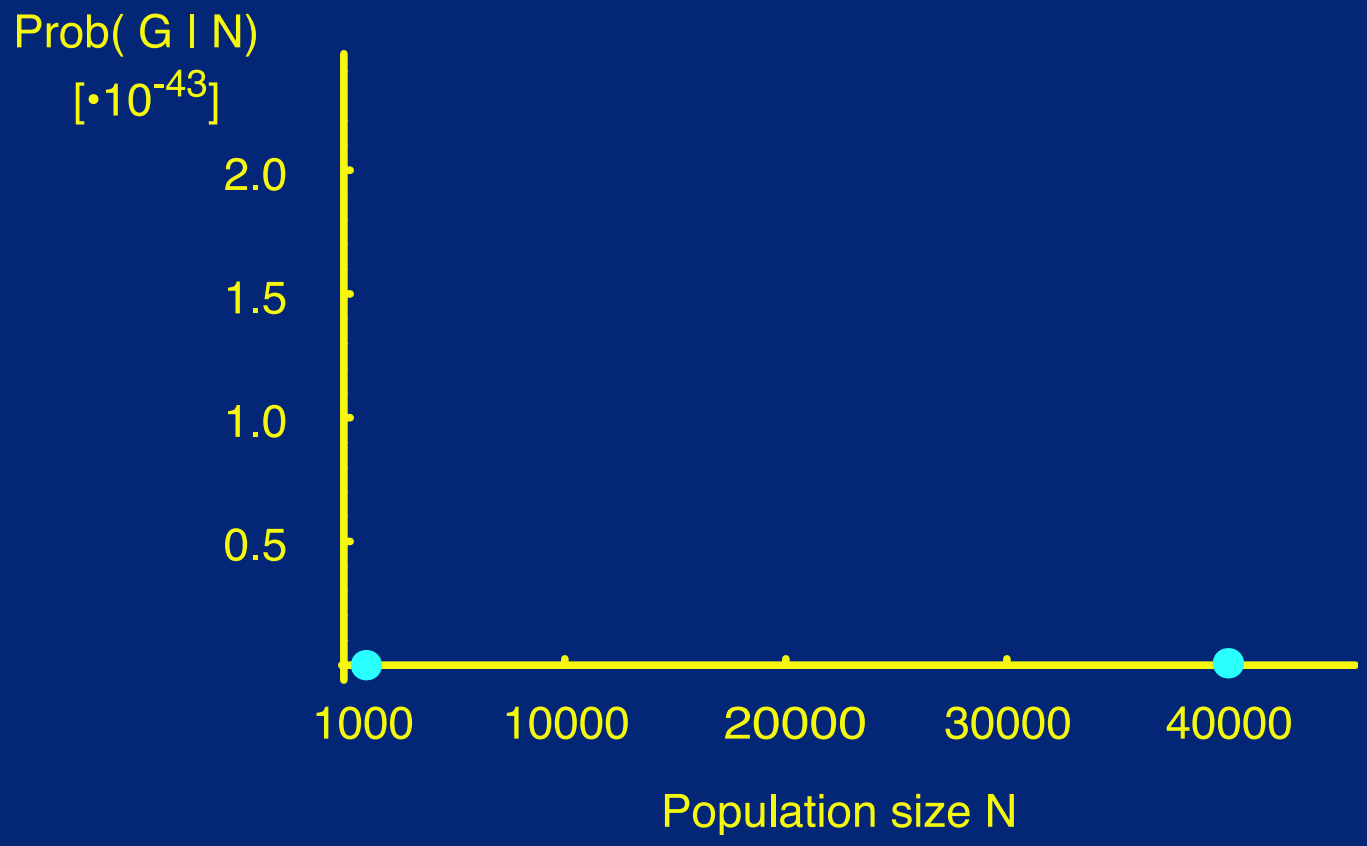
2. Calculate the size of the population



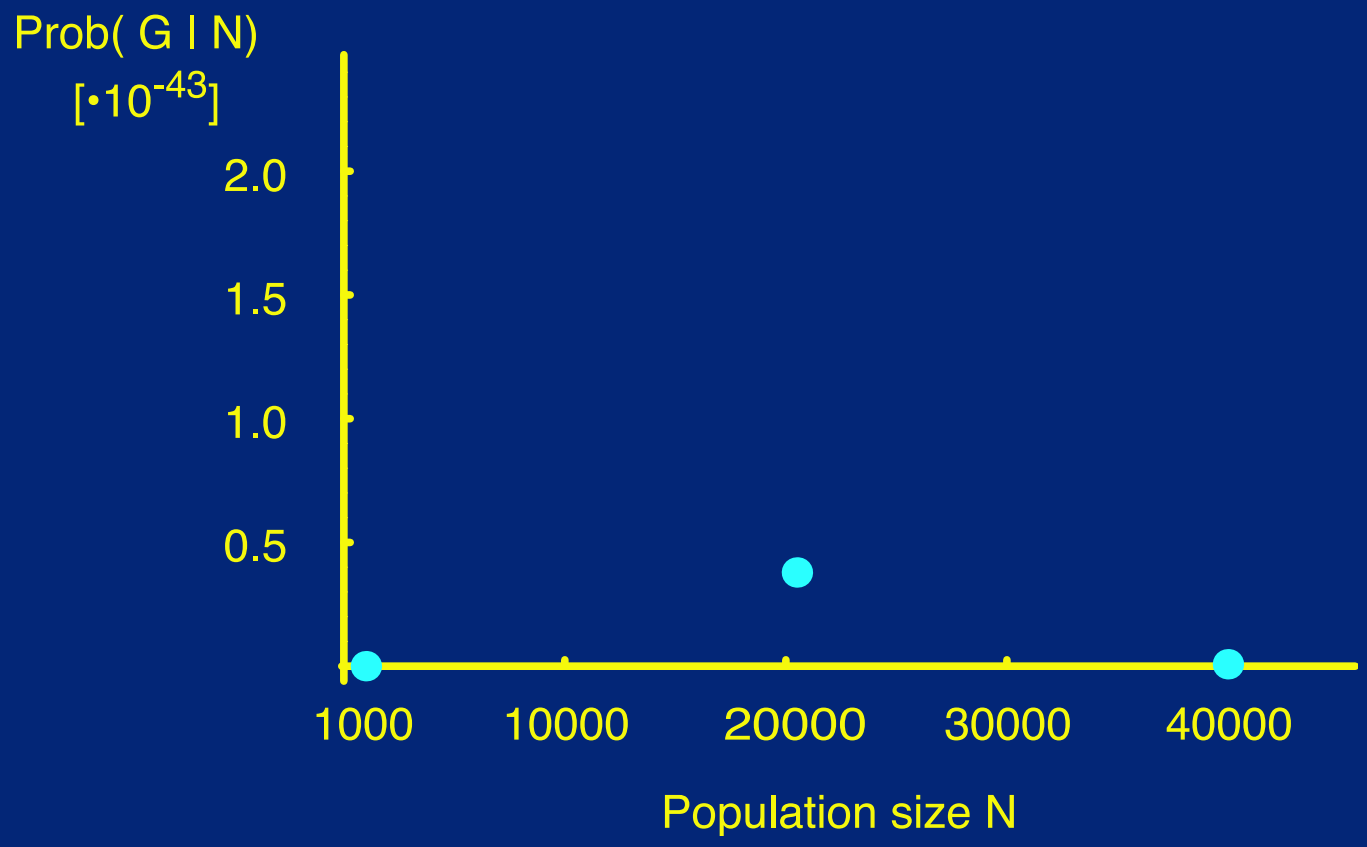
2. Calculate the size of the population



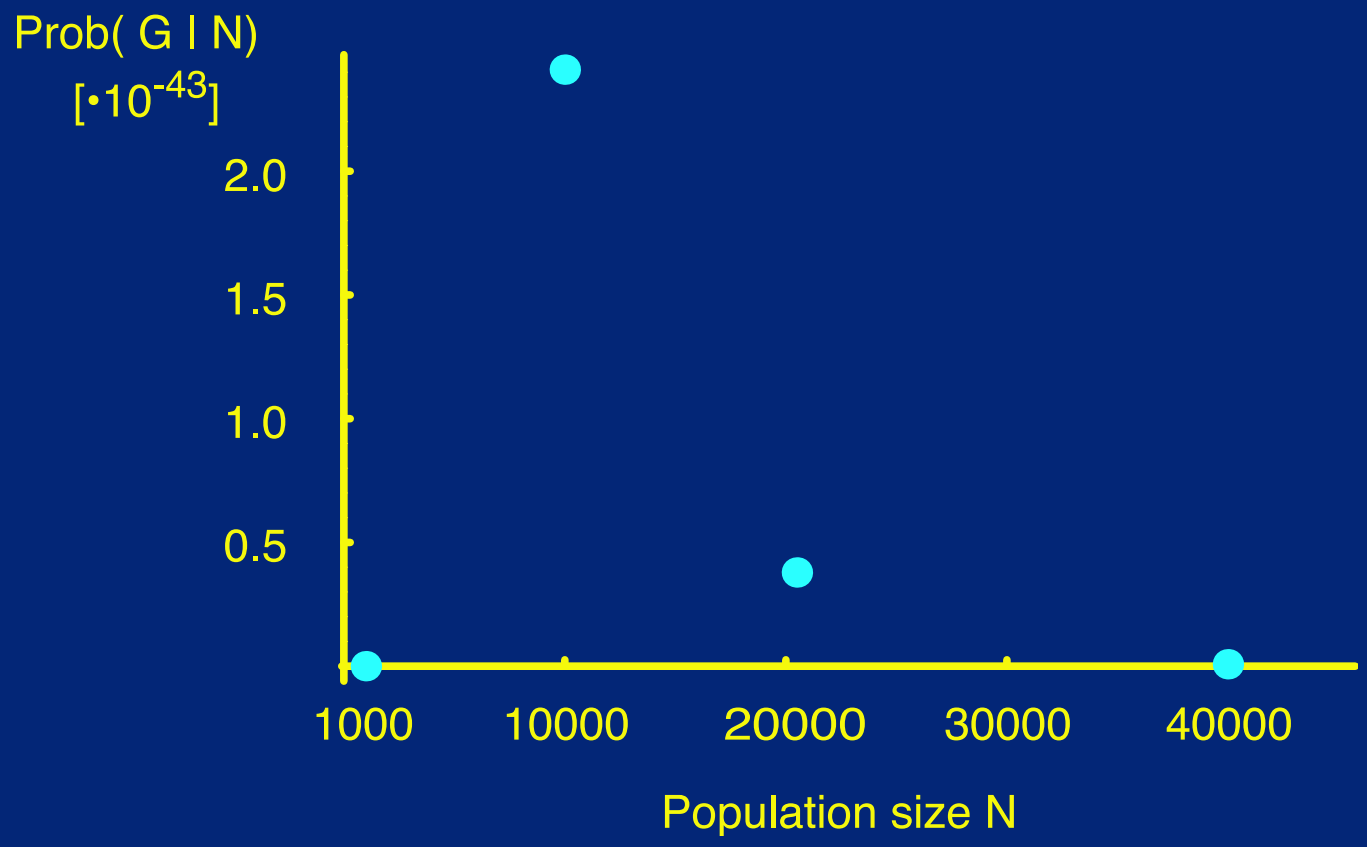
2. Calculate the size of the population



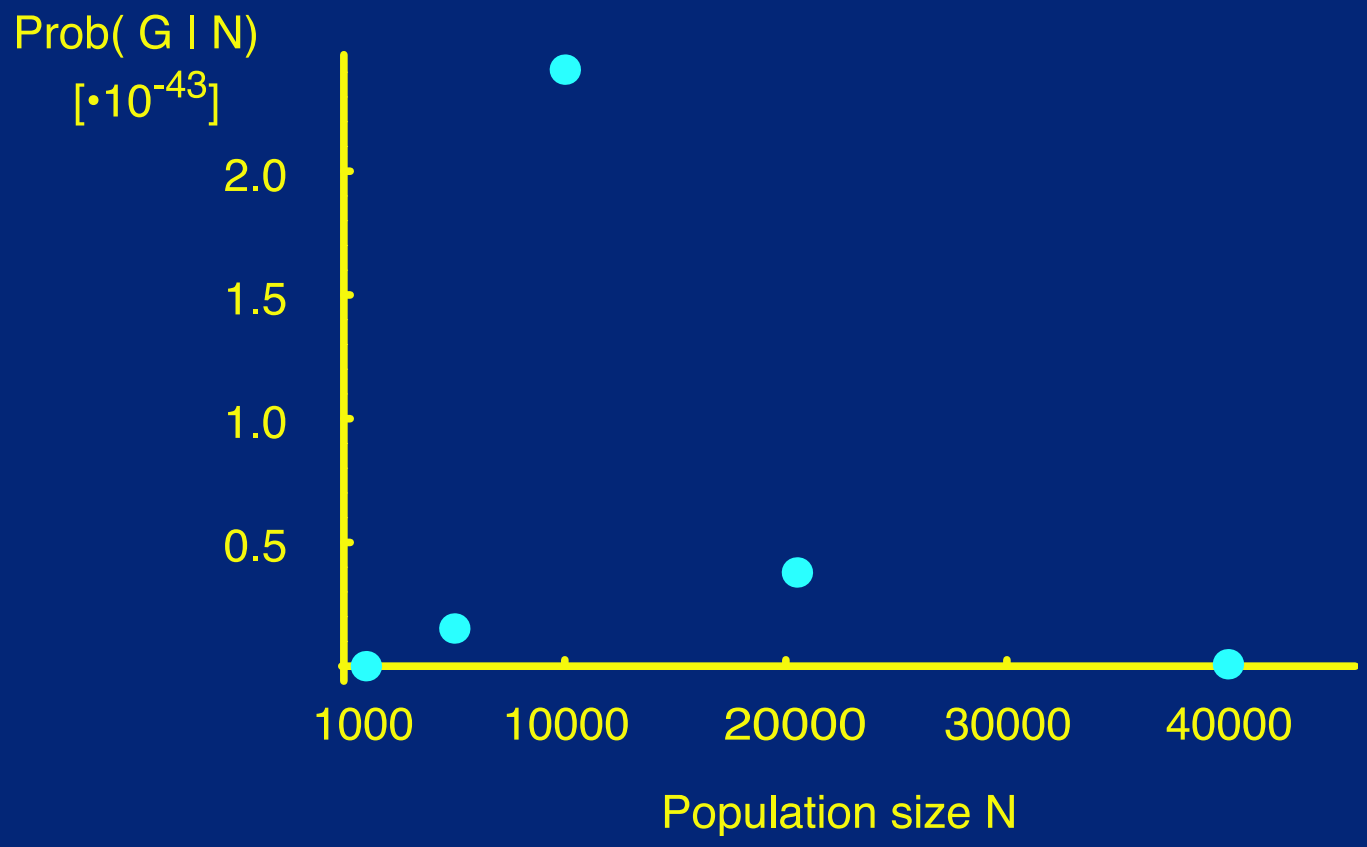
2. Calculate the size of the population



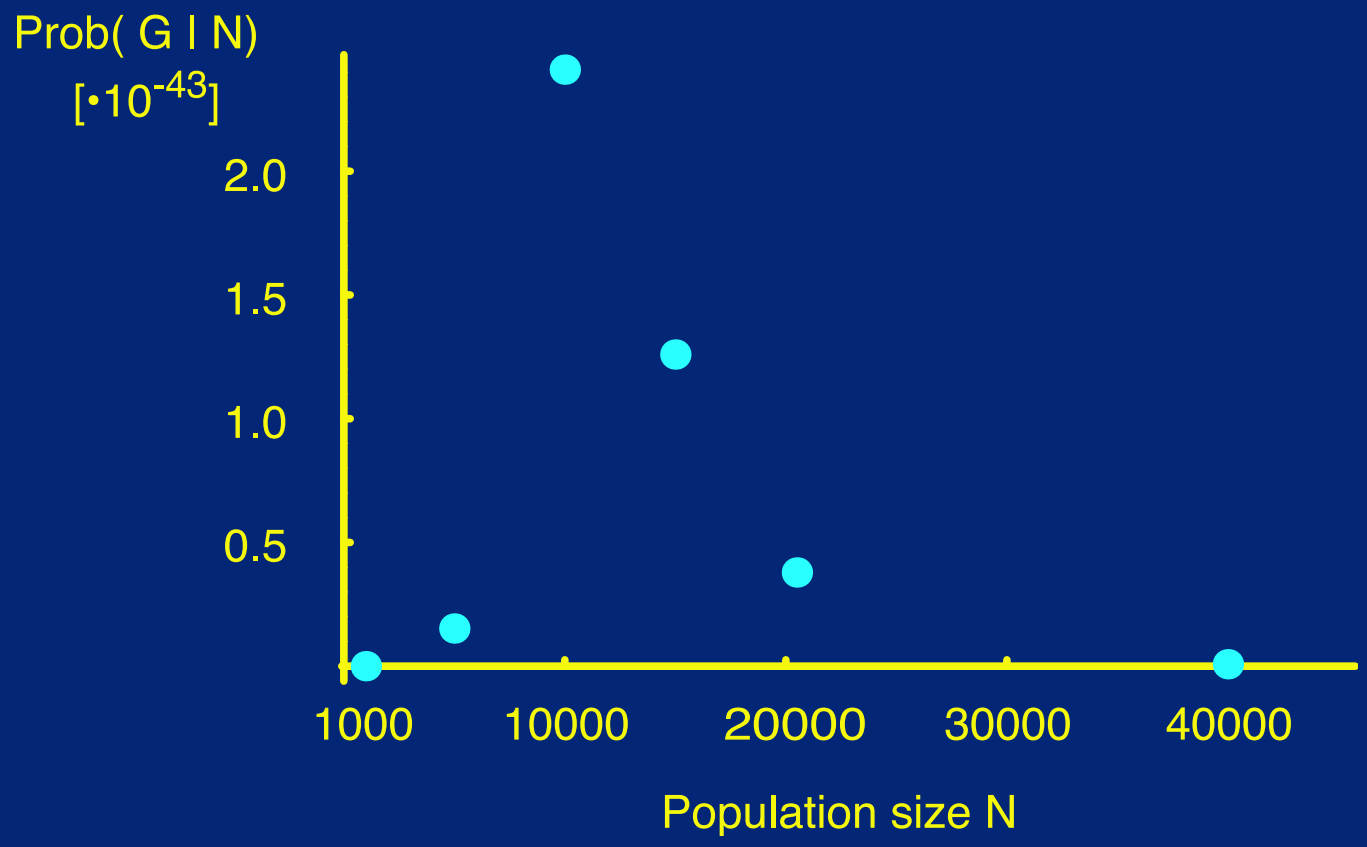
2. Calculate the size of the population



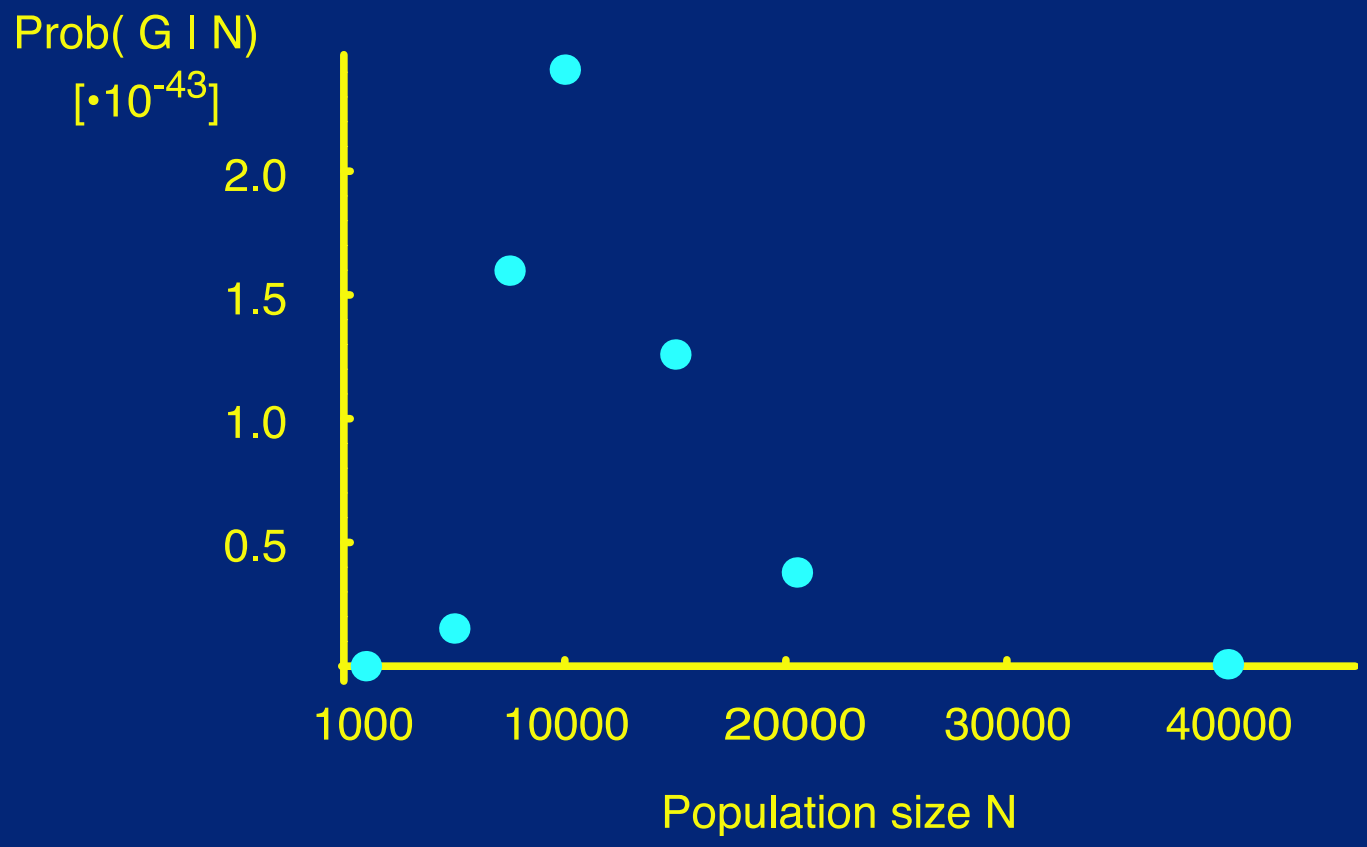
2. Calculate the size of the population



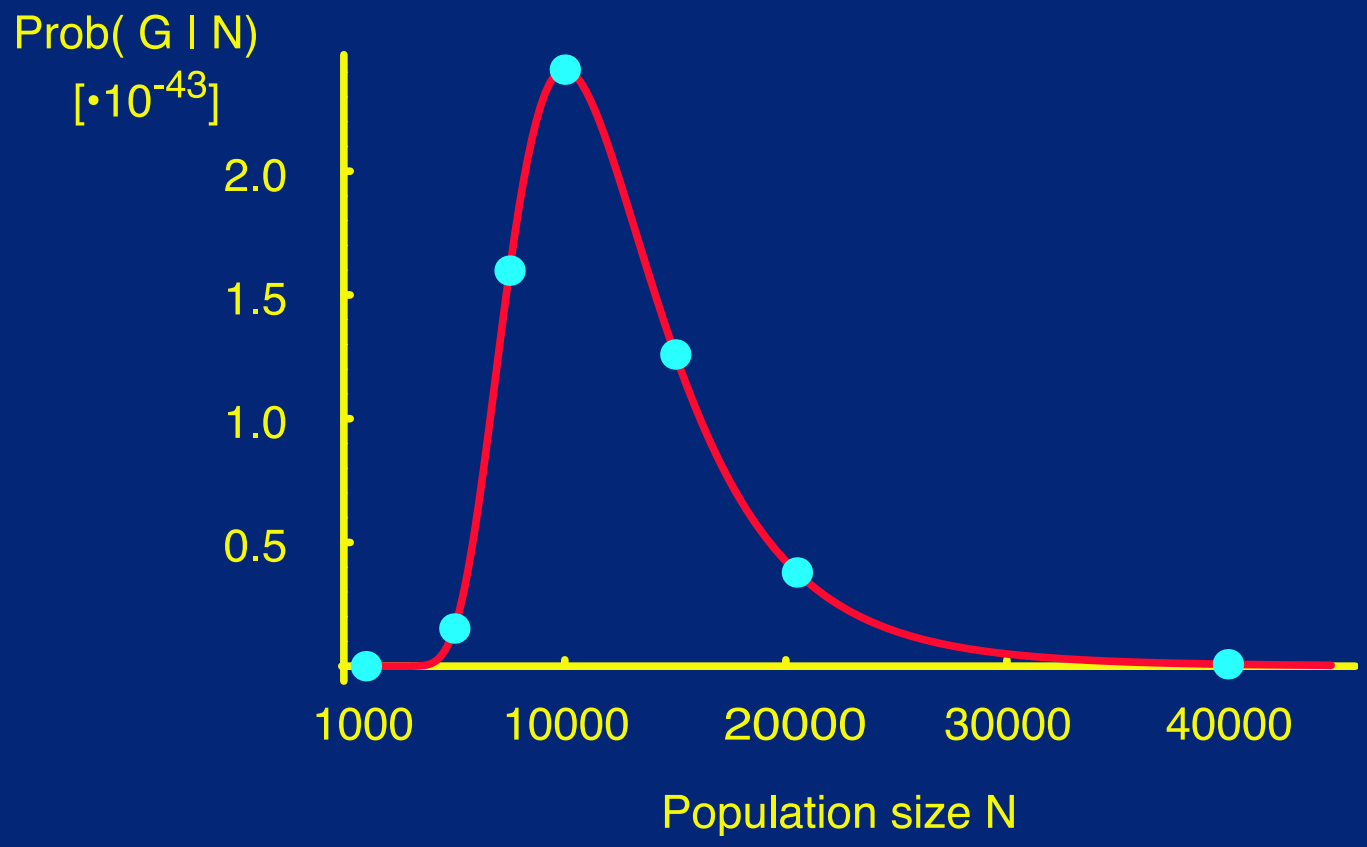
2. Calculate the size of the population



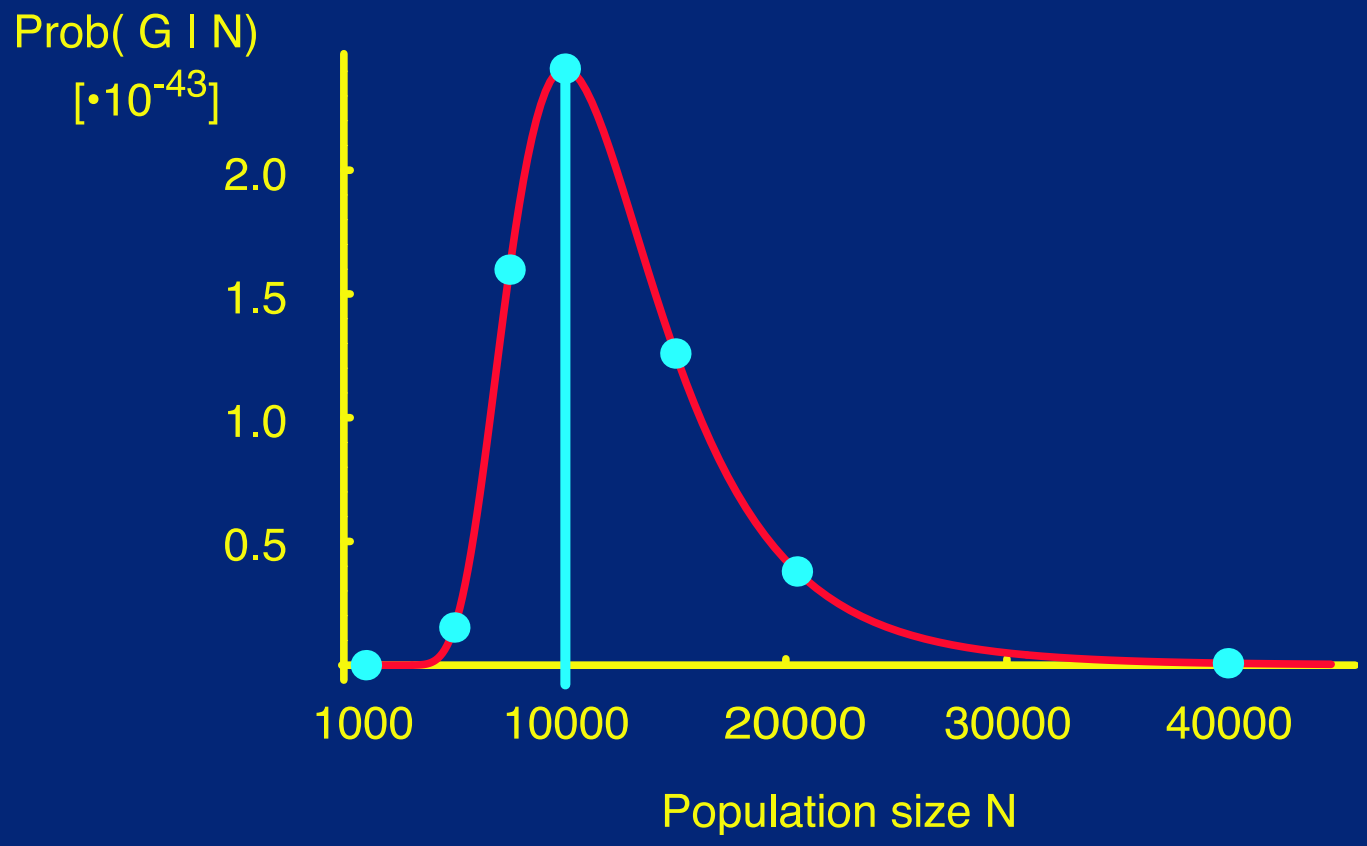
2. Calculate the size of the population



2. Calculate the size of the population

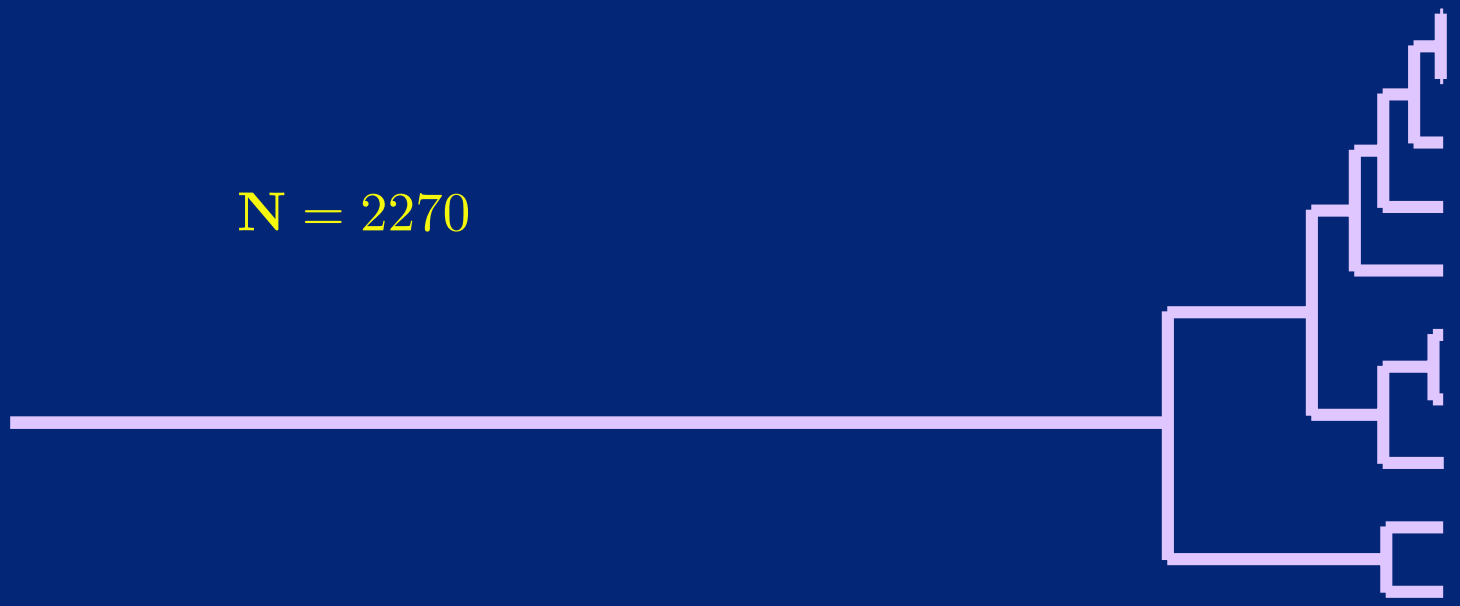


2. Calculate the size of the population

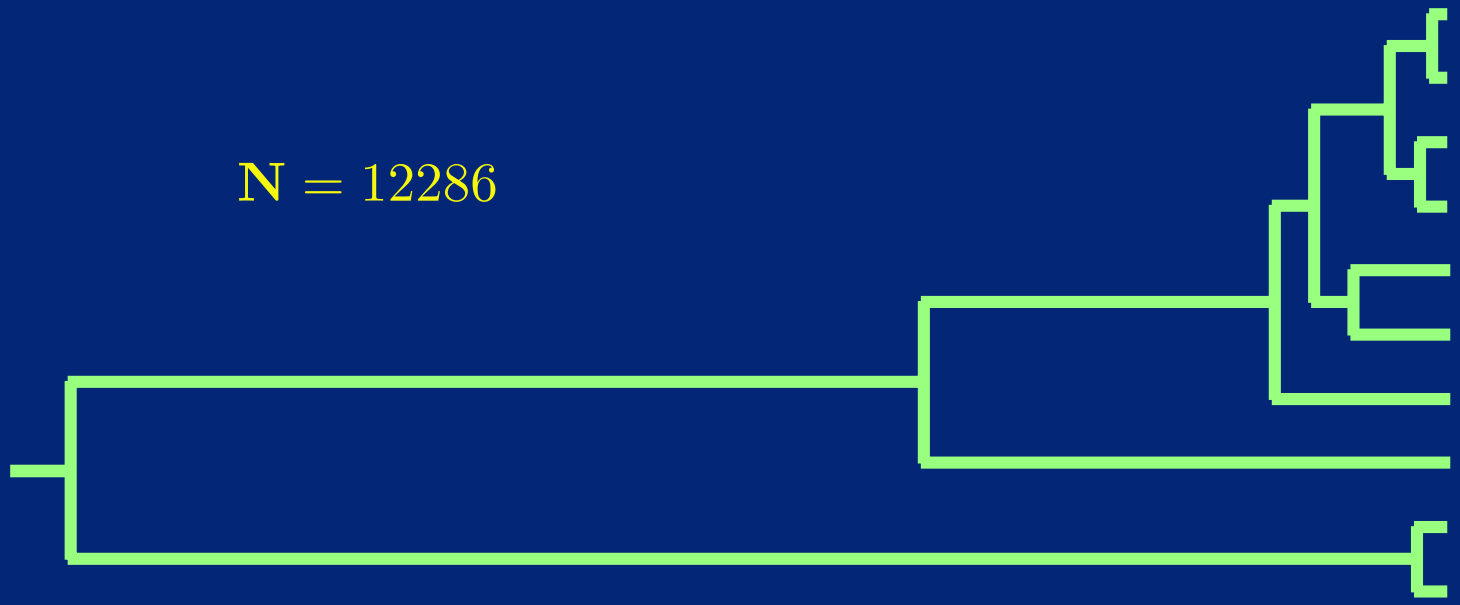


2. Calculate the size of the population

$\mathbf{N} = 2270$



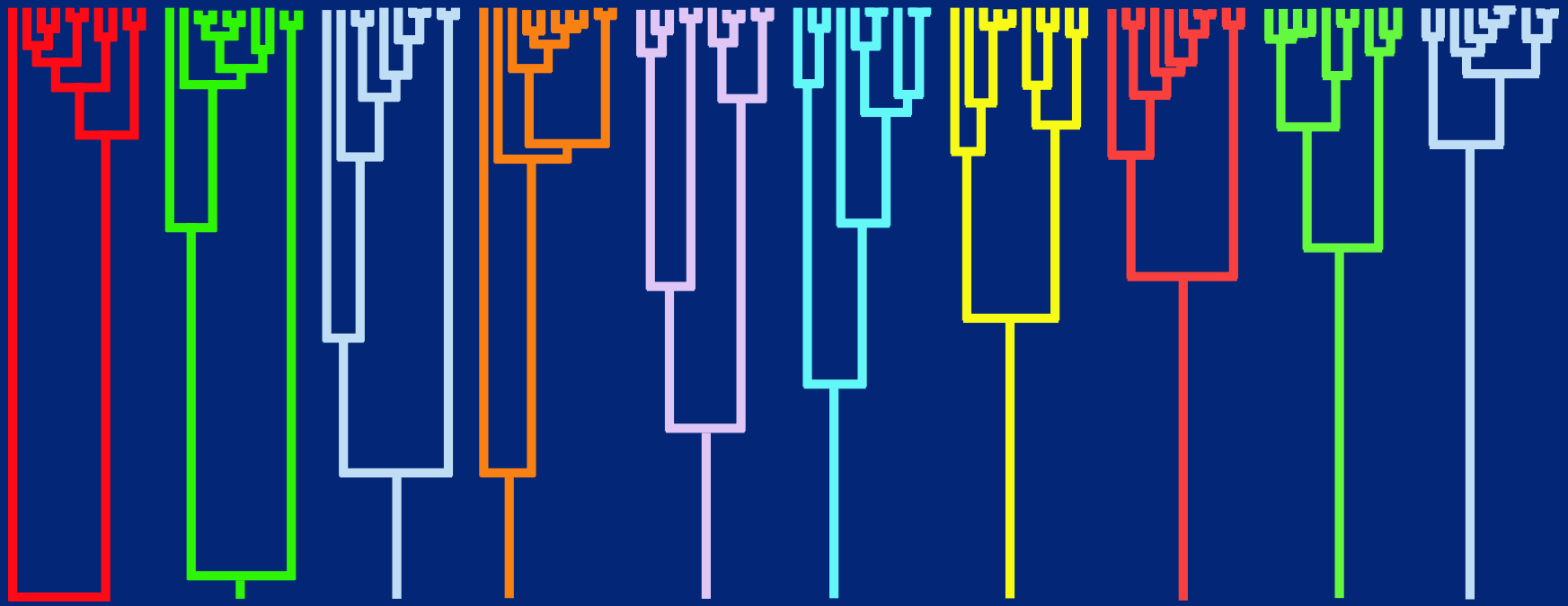
$\mathbf{N} = 12286$



Problems with these very naive approaches

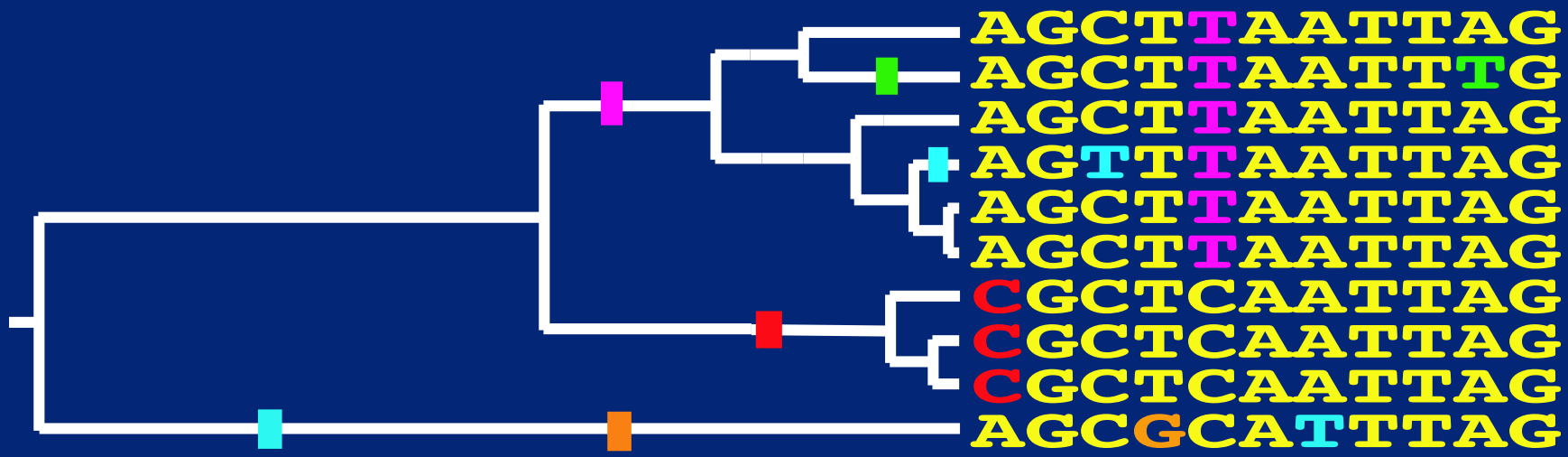
We assume we know the TRUE genealogy: topology and branch length.

Variability of the coalescent

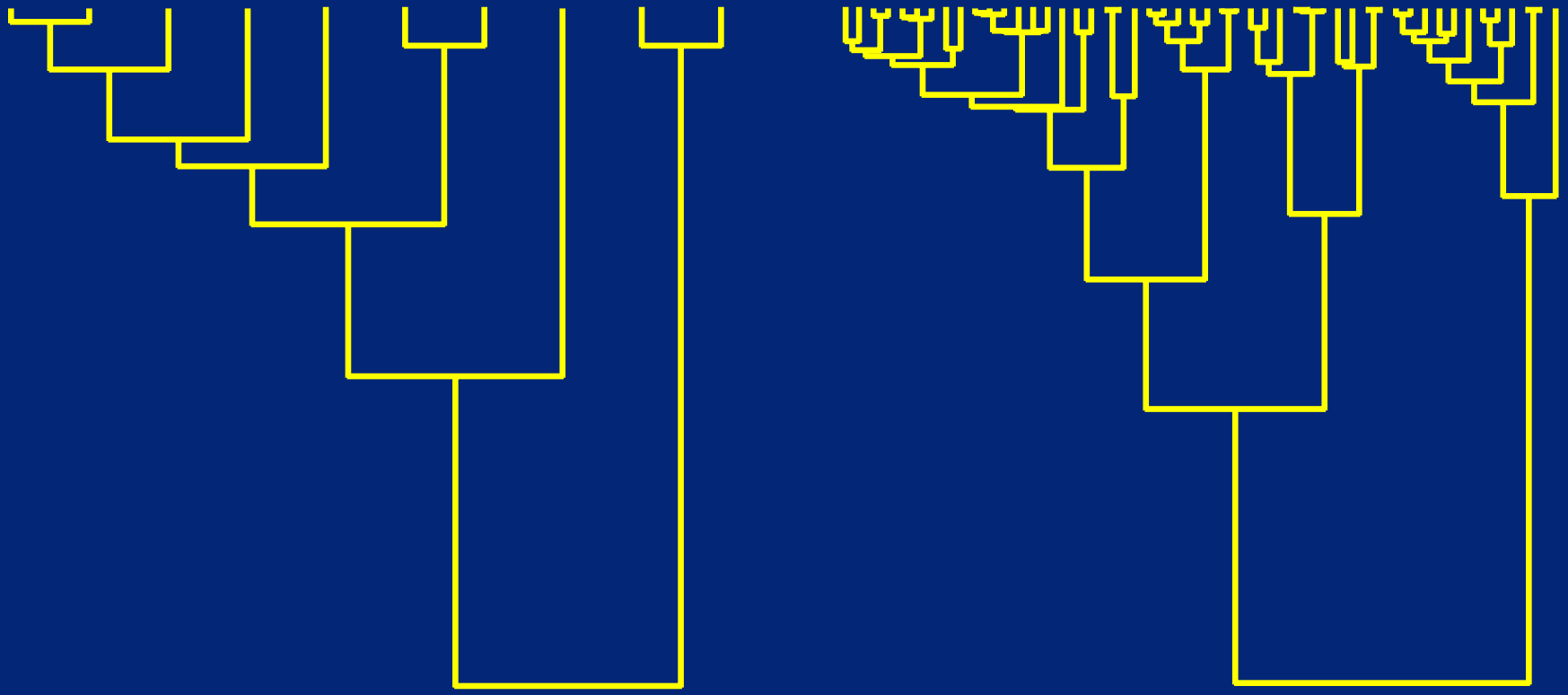


10 coalescent trees generated with the same population size, $N = 10,000$

Variability of mutations



How many samples do we need?



Summary of the basic *Coalescent*

- Mathematically tractable way to calculate probabilities of genealogies in a population.
- The coalescent is a very noisy distribution of times on a tree.
- Variability because of mutation increases the uncertainty of these times.
- The population size is correlated with the depth of the tree.
- Estimations of population size or the time of the MRCA from a single tree are very error-prone.

Parameter estimation using *maximum likelihood*

- Mutation model: Nucleotide mutation model, ...
- Population genetics model: the Coalescent

Parameter estimation using *maximum likelihood*

- Mutation model: Nucleotide mutation model, ...
- Population genetics model: the Coalescent

$$\text{Prob}\ (\mathbf{N}, \mu | \text{data})$$

Parameter estimation using *maximum likelihood*

- Mutation model: Nucleotide mutation model, ...
- Population genetics model: the Coalescent

$$\text{Prob } (\mathbf{N}, \mu|\text{data}) = \frac{\text{Prob } (\text{data}|\mathbf{N}, \mu) \text{ Prob } (\mathbf{N}, \mu)}{\text{Prob } (\text{data})}$$

Parameter estimation using *maximum likelihood*

- Mutation model: Nucleotide mutation model, ...
- Population genetics model: the Coalescent

$$\text{Prob } (\mathbf{N}, \mu|\text{data}) = \frac{\text{Prob } (\text{data}|\mathbf{N}, \mu) \text{ Prob } (\mathbf{N}, \mu)}{\text{Prob } (\text{data})}$$

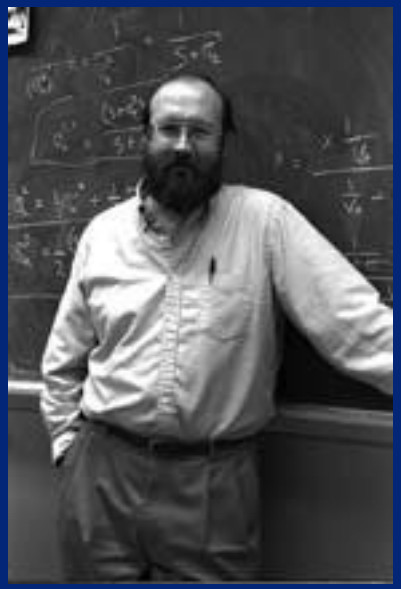
$$\text{L}(\mathbf{N}, \mu) = \text{Prob } (\text{data}|\mathbf{N}, \mu) = \ c \text{ Prob } (\mathbf{N}, \mu|\text{data})$$

Parameter estimation using *maximum likelihood*

$$\mathrm{L}(\mathbf{N}, \mu) = \mathrm{Prob}\ (\mathsf{data}|\mathbf{N}, \mu)$$

Parameter estimation using *maximum likelihood*

$$\mathrm{L}(\mathbf{N},\mu) = \mathrm{Prob}\ (\mathsf{data}|\mathbf{N},\mu) = \int_G \mathrm{p}(\mathrm{G}|\mathbf{N},\mu)\ \mathrm{Prob}\ (\mathsf{data}|\mathrm{G},\mu)$$



Parameter estimation using *maximum likelihood*

$$\mathrm{L}(\mathbf{N}, \mu) = \mathrm{Prob}\ (\mathrm{data}|\mathbf{N}, \mu) = \int_G \mathrm{p}(\mathrm{G}|\mathbf{N}, \mu)\ \mathrm{Prob}\ (\mathrm{data}|\mathrm{G}, \mu)$$

We cannot observe the mutation events. Instead of estimating $\mathbf{N}$ and μ we estimate the product $\mathbf{\Theta} = 4\mathbf{N}\mu$ and scale G with μ

$$\mathrm{L}(\mathbf{\Theta}) = \int_{G^*} \mathrm{p}(\mathrm{G}^*|\mathbf{\Theta})\ \mathrm{Prob}\ (\mathrm{data}|\mathrm{G}^*)$$

Parameter estimation using *maximum likelihood*

$$\mathrm{L}(\mathbf{N}, \mu) = \mathrm{Prob}\ (\mathrm{data}|\mathbf{N}, \mu) = \int_G \mathrm{p}(\mathrm{G}|\mathbf{N}, \mu)\ \mathrm{Prob}\ (\mathrm{data}|\mathrm{G}, \mu)$$

We cannot observe the mutation events. Instead of estimating $\mathbf{N}$, and μ we estimate the product $\mathbf{\Theta} = 4\mathbf{N}\mu$ and scale G with μ

$$\mathrm{L}(\mathbf{\Theta}) = \int_{G^*} \mathrm{p}(\mathrm{G}^*|\mathbf{\Theta})\ \mathrm{Prob}\ (\mathrm{data}|\mathrm{G}^*)$$

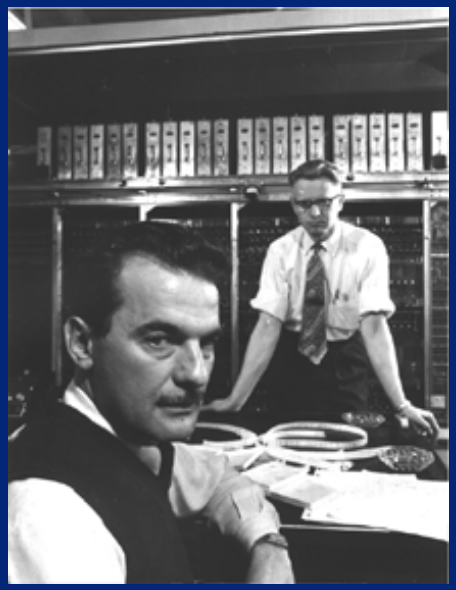
Problem: We need to integrate over all genealogies: all different labelled histories, all different branchlengths

Can we calculate this sum over all genealogies?

We need to integrate over all genealogies: all different topologies, all different branchlengths

Tips	Topologies
3	3
4	18
5	180
6	2700
7	56700
8	1587600
9	57153600
10	2571912000
15	69580576689624000000
20	564480989588730591336960000000
30	436846661310306951246468019862076389144064000000000000
40	302733382994800735654630336455145720004293943205386250170788872192000000000000000000
50	3.28632×10^{112}
100	1.37416×10^{284}

A solution: *Markov chain Monte Carlo*



Metropolis recipe

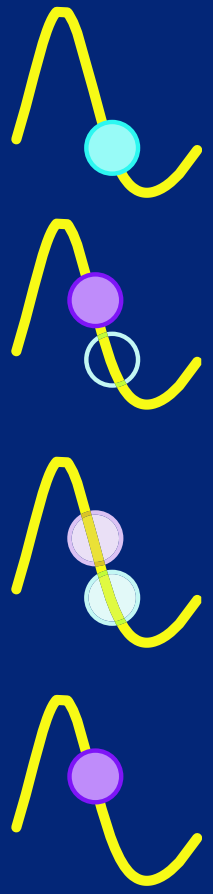
0. first state

1. perturb old state and calculate probability of new state

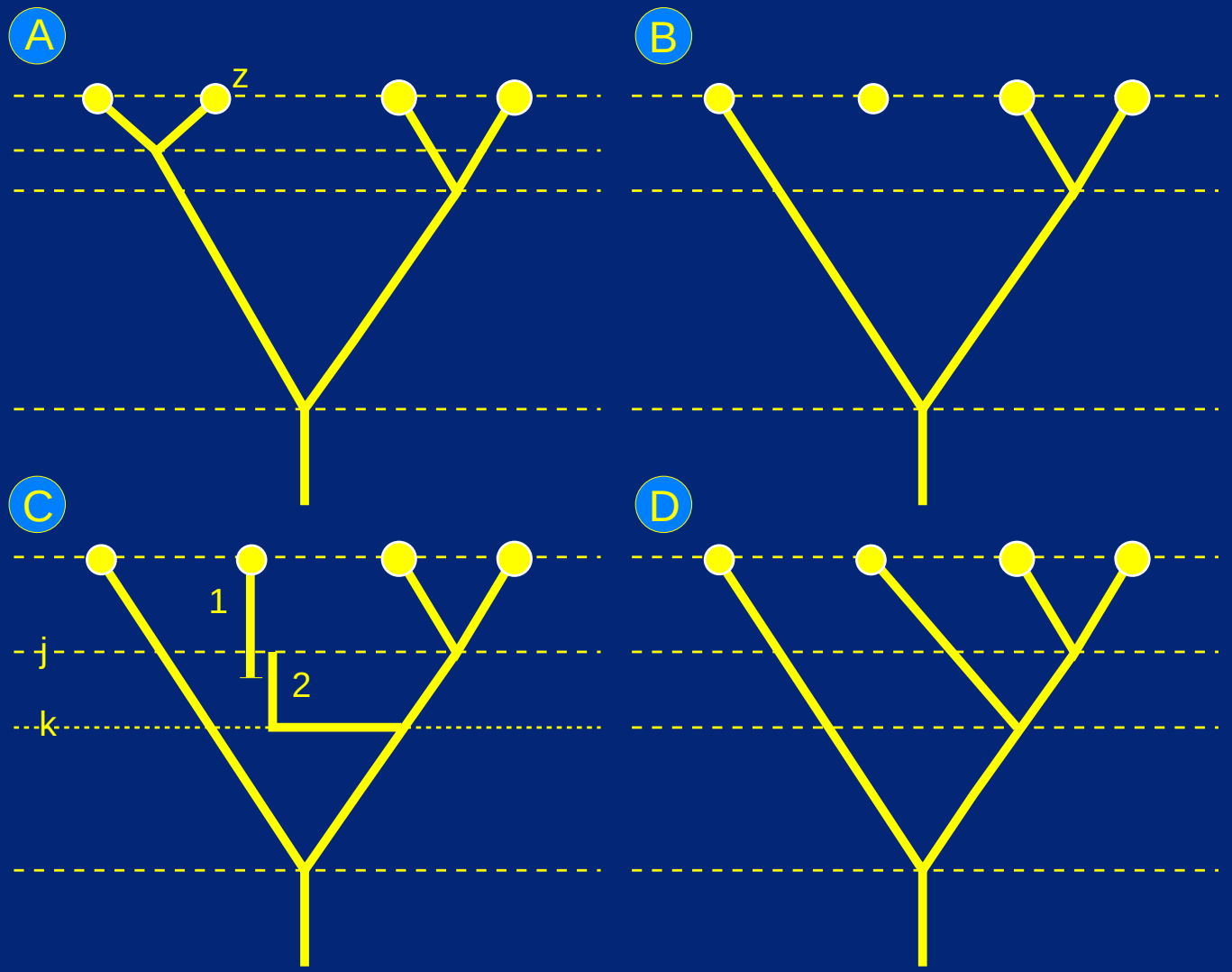
2. test if new state is better than old state: accept if ratio of new and old is larger than a random number between 0 and 1.

3. move to new state if accepted otherwise stay at old state

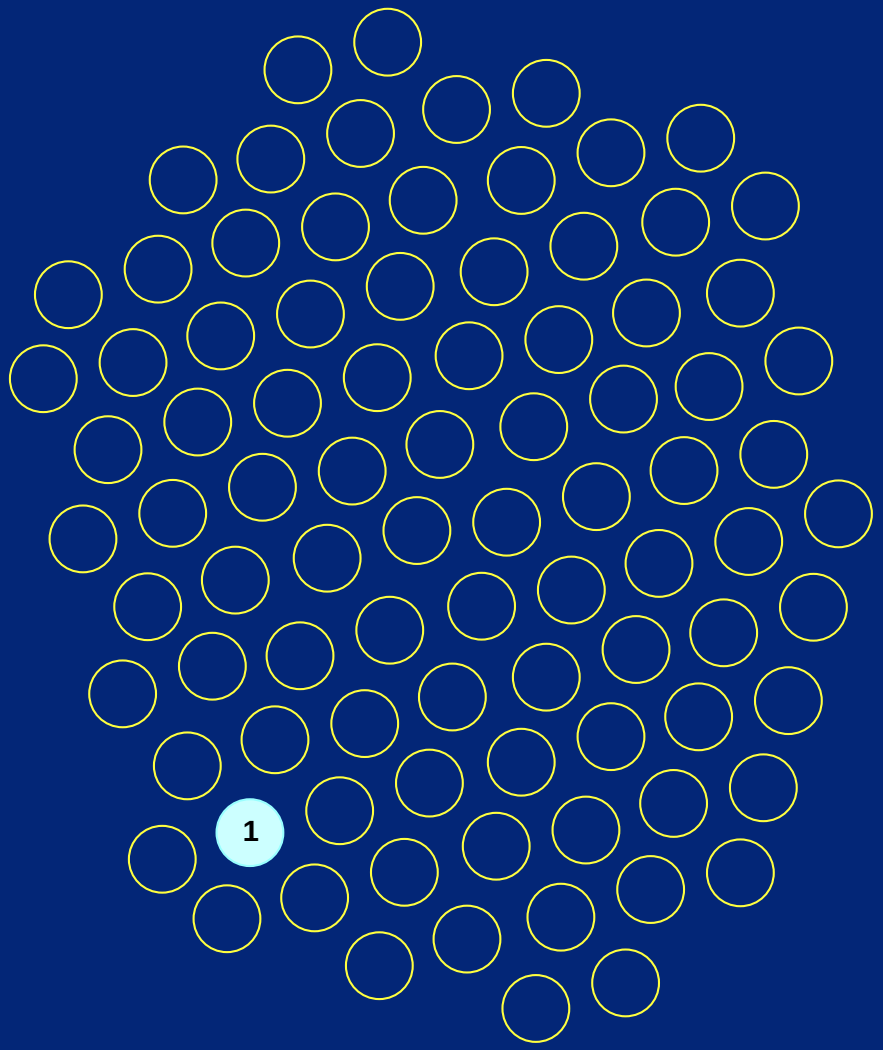
4. go to 1



How do we change a genealogy

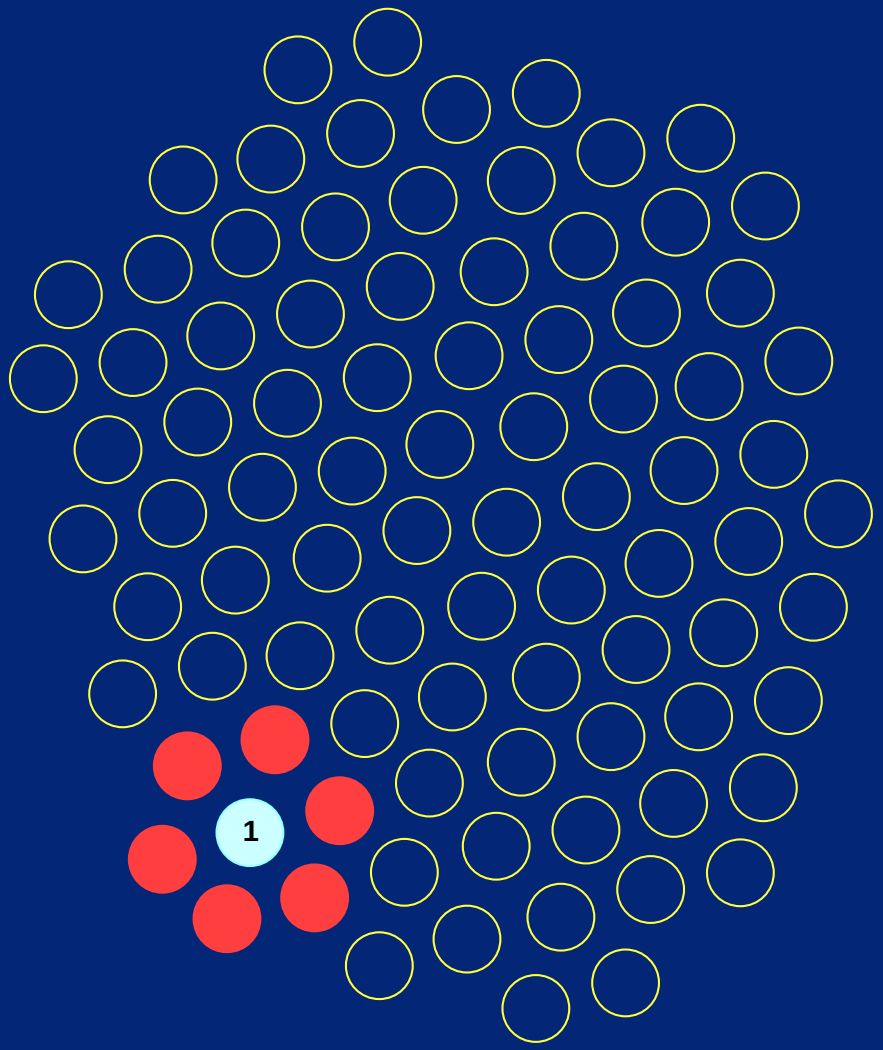


Markov chain Monte Carlo



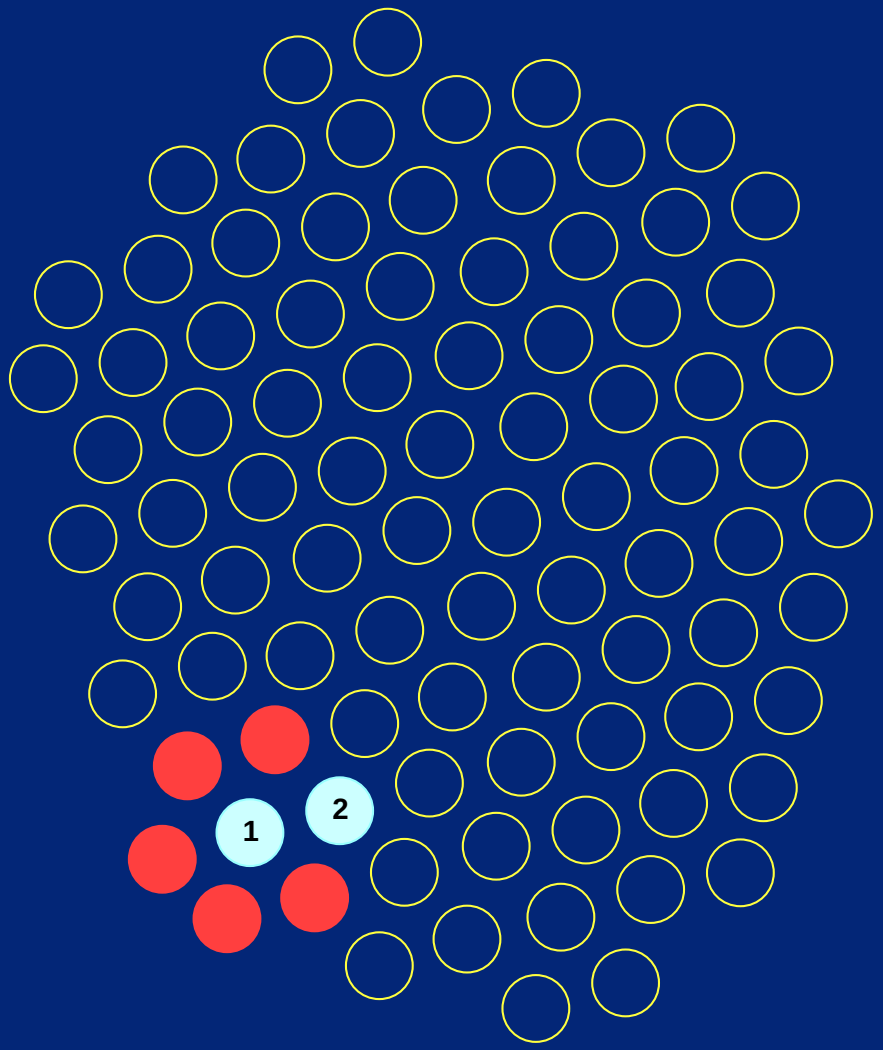
$\mathrm{L}(G_1|\Theta)$

Markov chain Monte Carlo



create a new tree

Markov chain Monte Carlo



$\mathrm{L}(G_2|\Theta)$

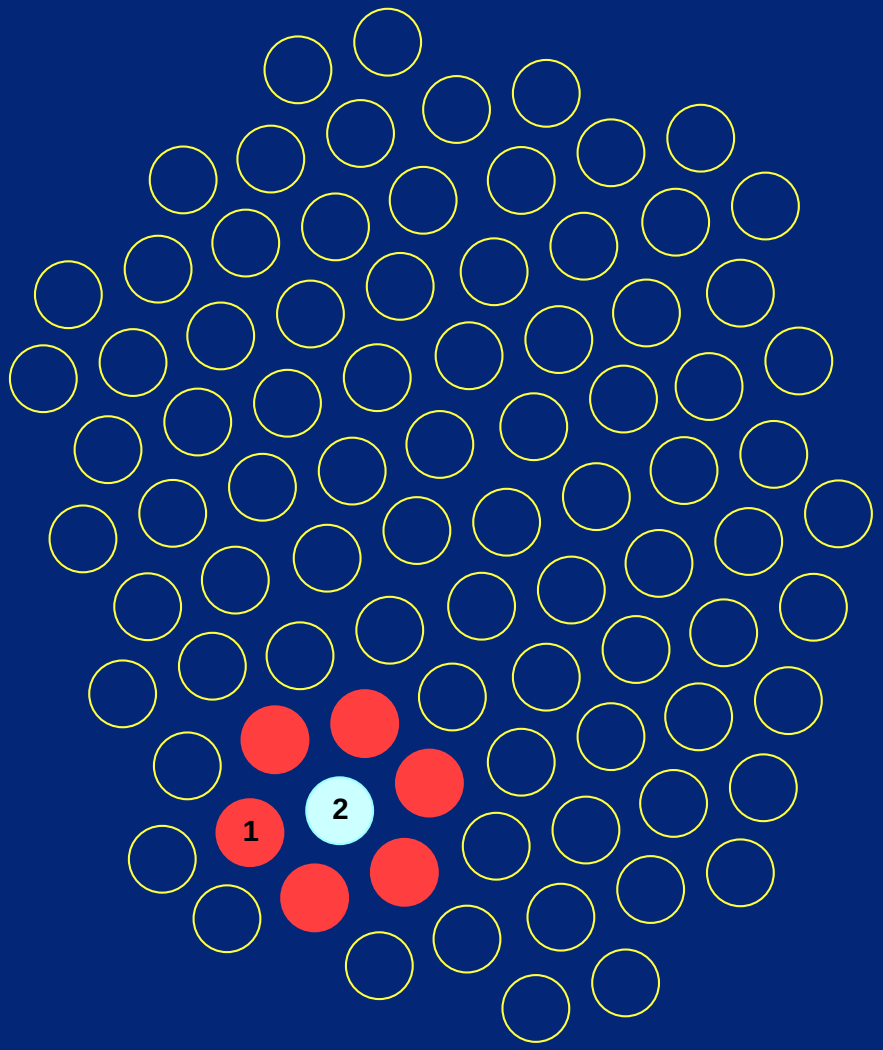
Evaluate

$$r < \frac{\mathrm{p}(\mathrm{G}_2|\Theta)\mathrm{P}(\mathrm{D}|\mathrm{G}_2)\mathrm{P}(\mathrm{G}_1|\mathrm{G}_2)}{\mathrm{p}(\mathrm{G}_1|\Theta)\mathrm{P}(\mathrm{D}|\mathrm{G}_1)\mathrm{P}(\mathrm{G}_2|\mathrm{G}_1)}$$

luckily reduces most often to:

$$r < \frac{\mathrm{P}(\mathrm{D}|\mathrm{G}_2)}{\mathrm{P}(\mathrm{D}|\mathrm{G}_1)}$$

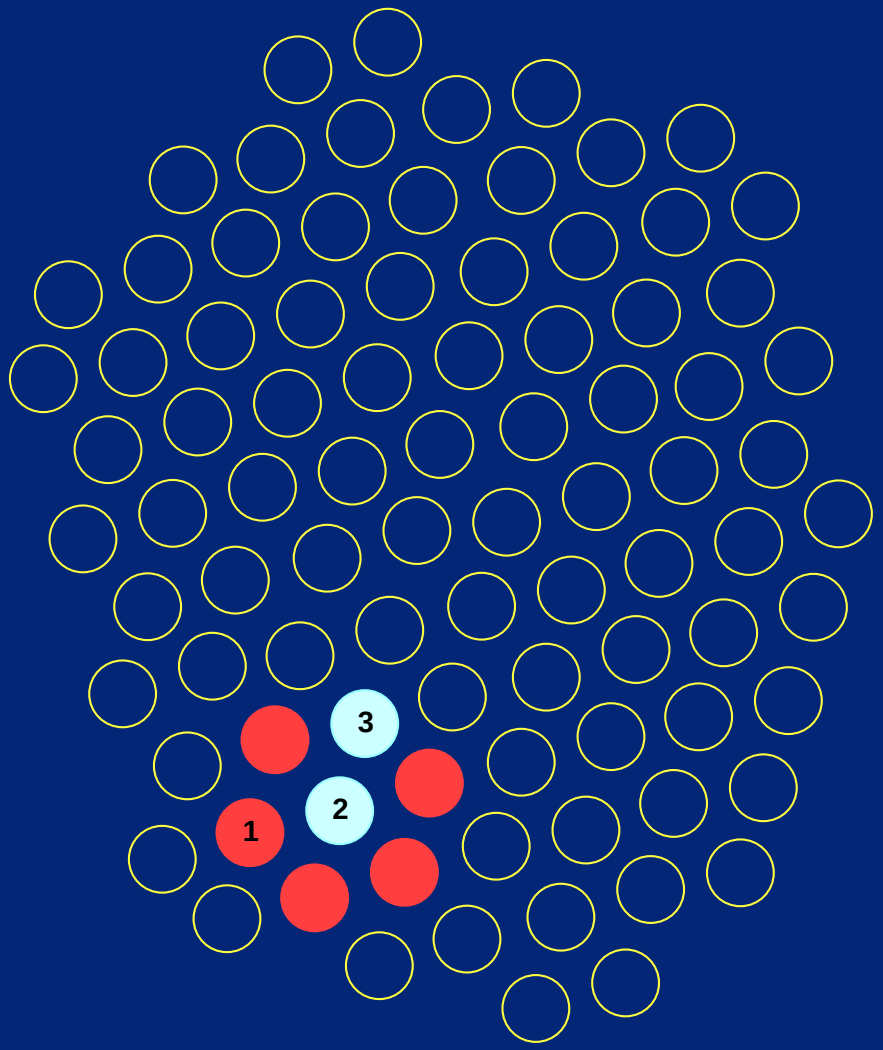
Markov chain Monte Carlo



Store G_1

Make another change to the tree

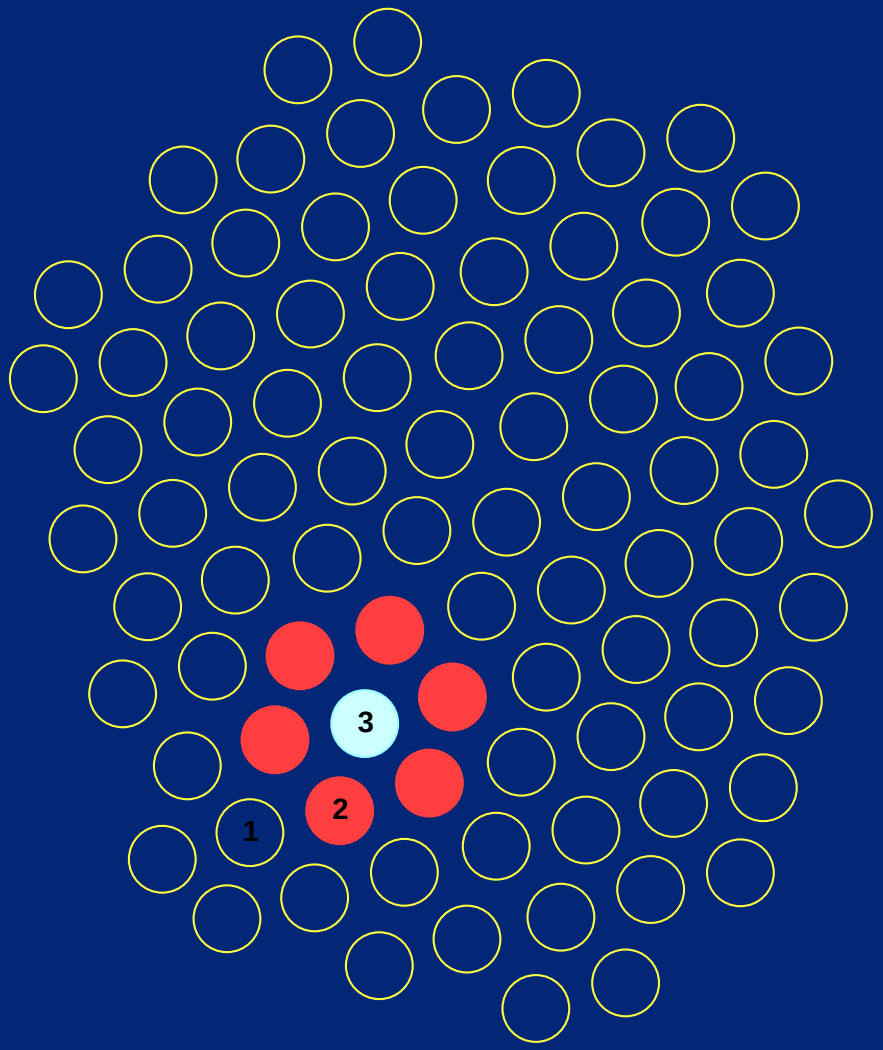
Markov chain Monte Carlo



$$\mathrm{L}(G_3|\Theta)$$

$$r < \frac{\mathrm{P}(\mathrm{D}|\mathrm{G}_3)}{\mathrm{P}(\mathrm{D}|\mathrm{G}_2)}$$

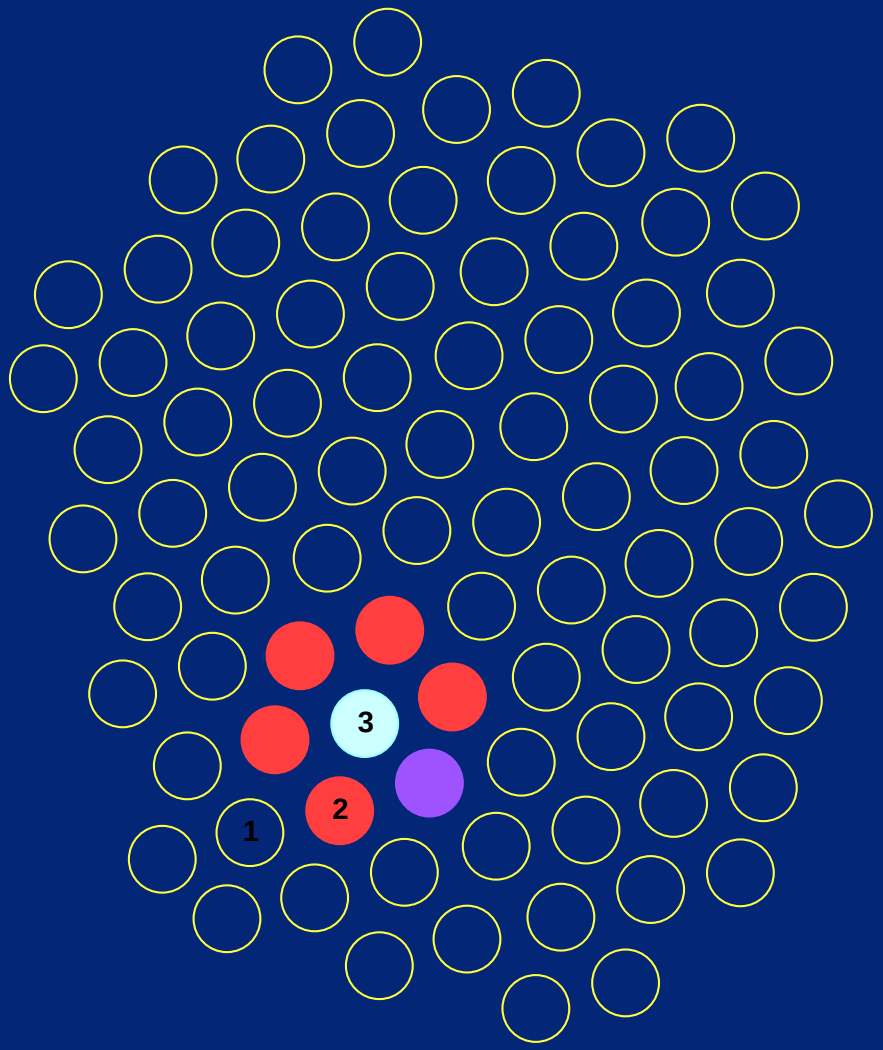
Markov chain Monte Carlo



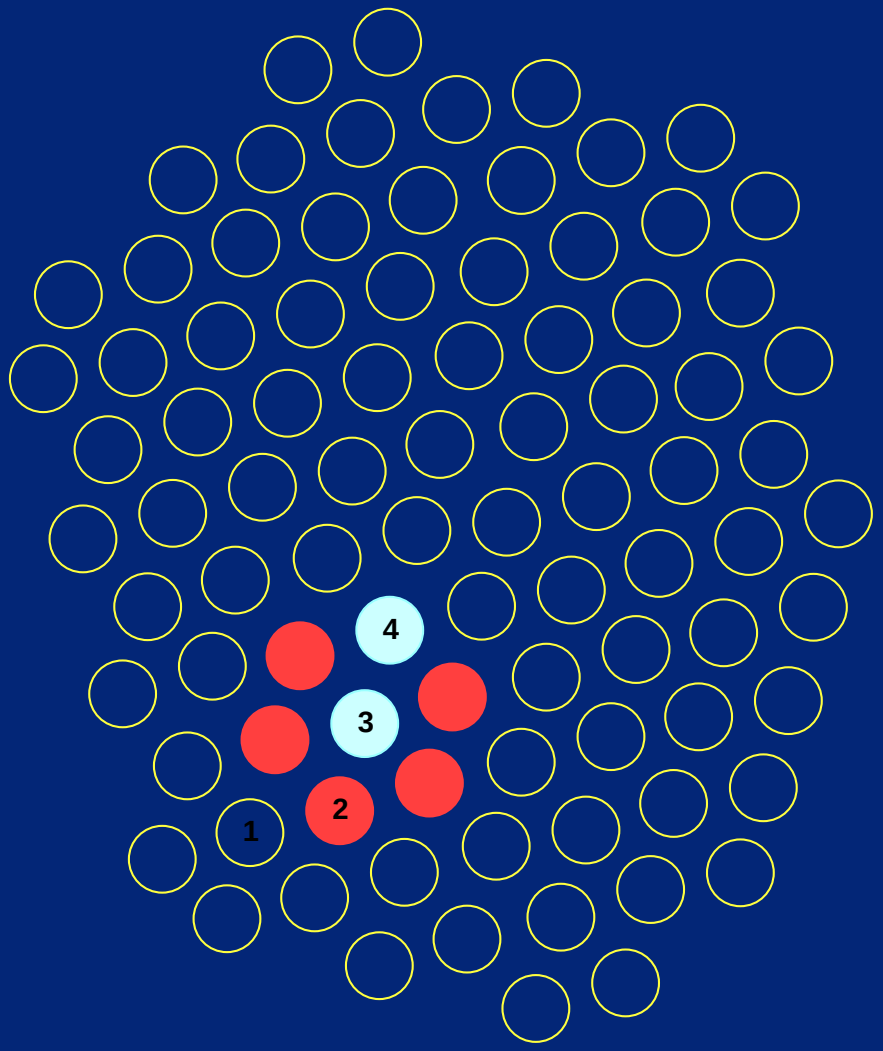
Store G_2

Make another change to the tree

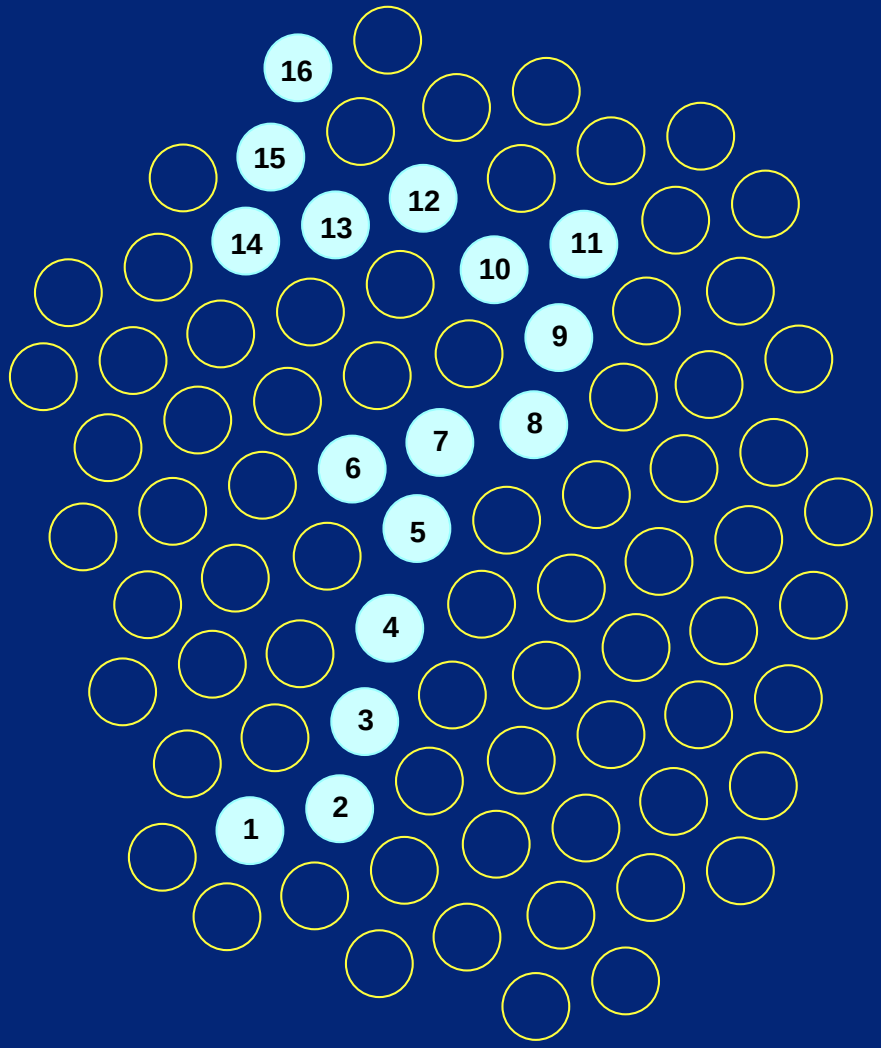
Markov chain Monte Carlo



Markov chain Monte Carlo

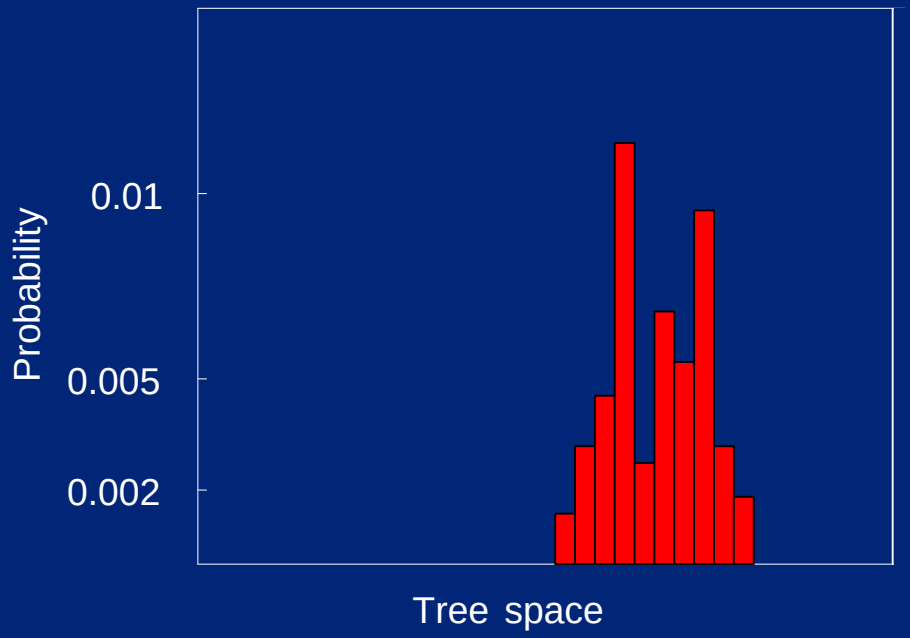


Markov chain Monte Carlo

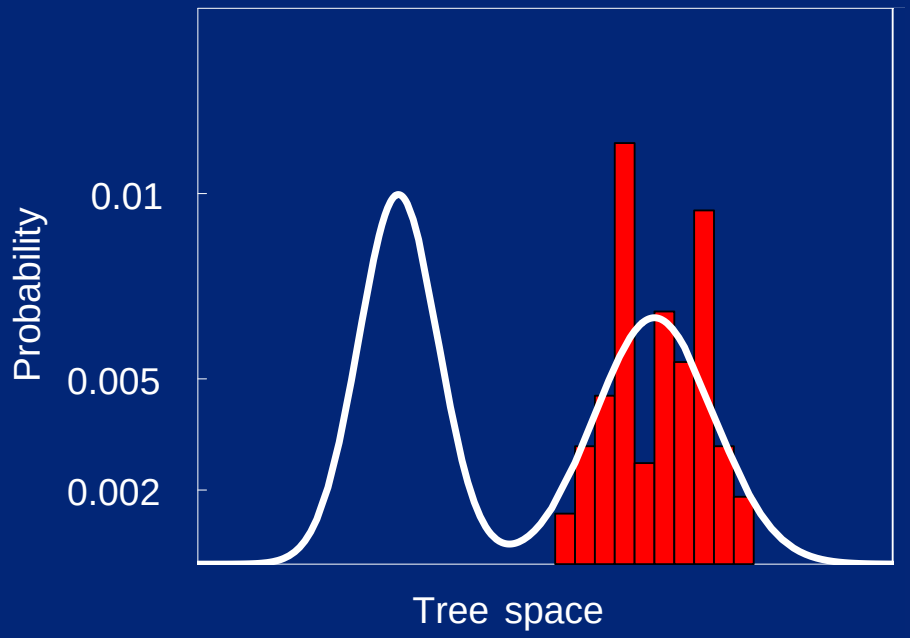


MCMC walk I

MCMC walk result



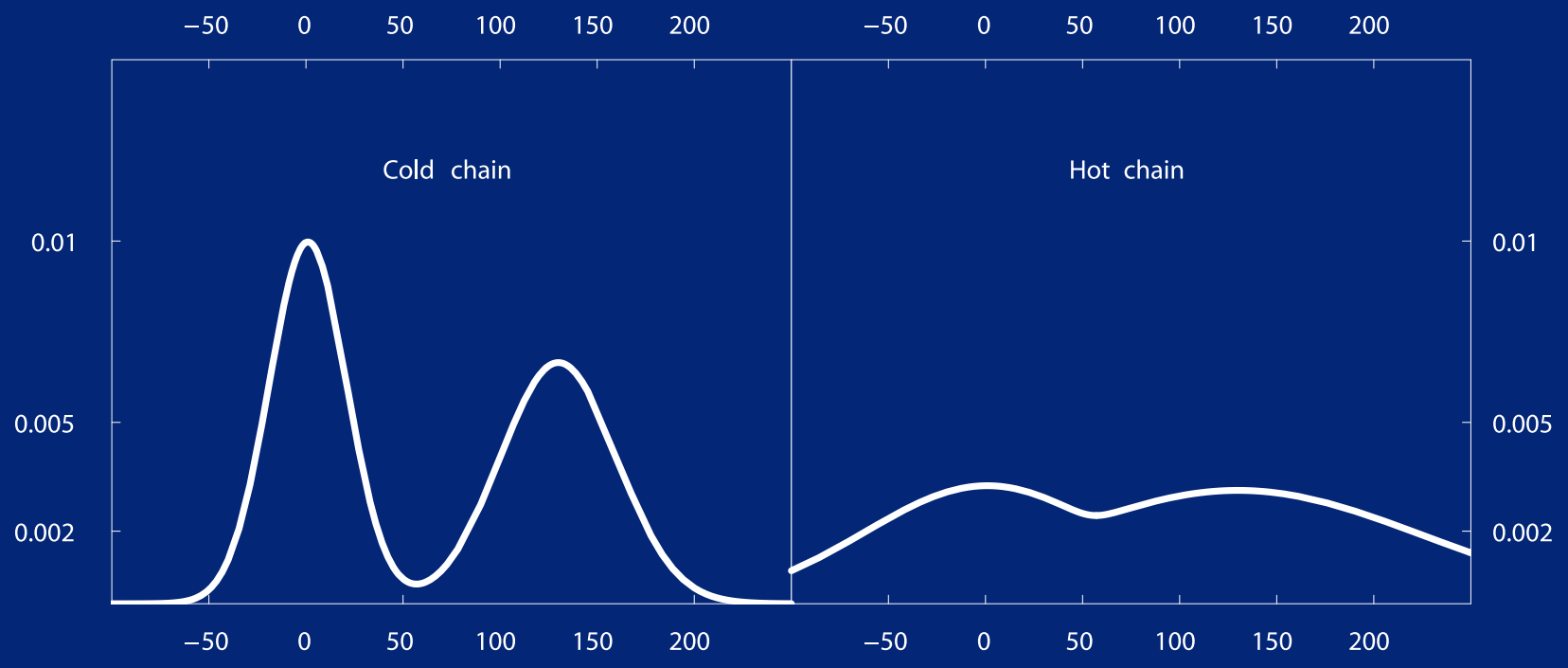
MCMC walk result



Improving our MCMC walker: MCMCMC or MC^3

Metropolis Coupled Markov chain Monte Carlo

- Run several independent parallel chains: each has a different temperature
- After some sampling of genealogies, swap the genealogies of a pair of chains if the ratio between probabilities in the cold and the hot chain is larger than a random number drawn between 0 and 1.



MCMC walk II

better MCMC walk result