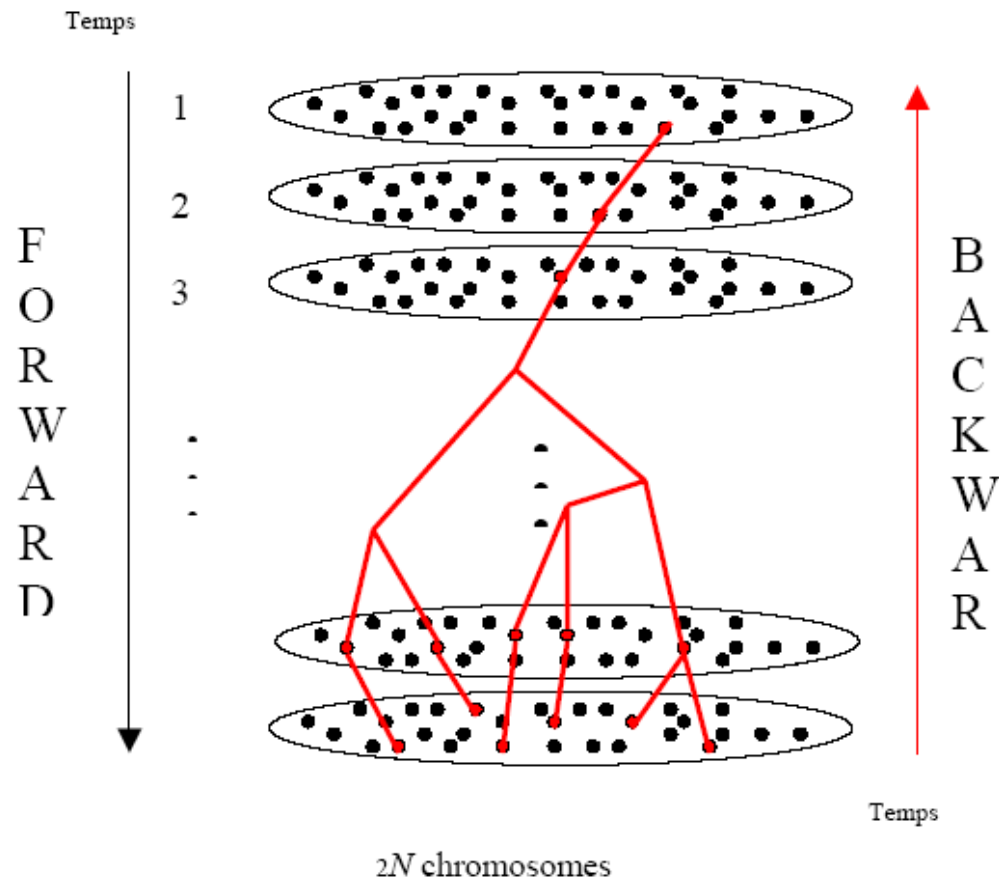


# Théorie de la coalescence, Simulation d'arbres et estimations de paramètres démographiques

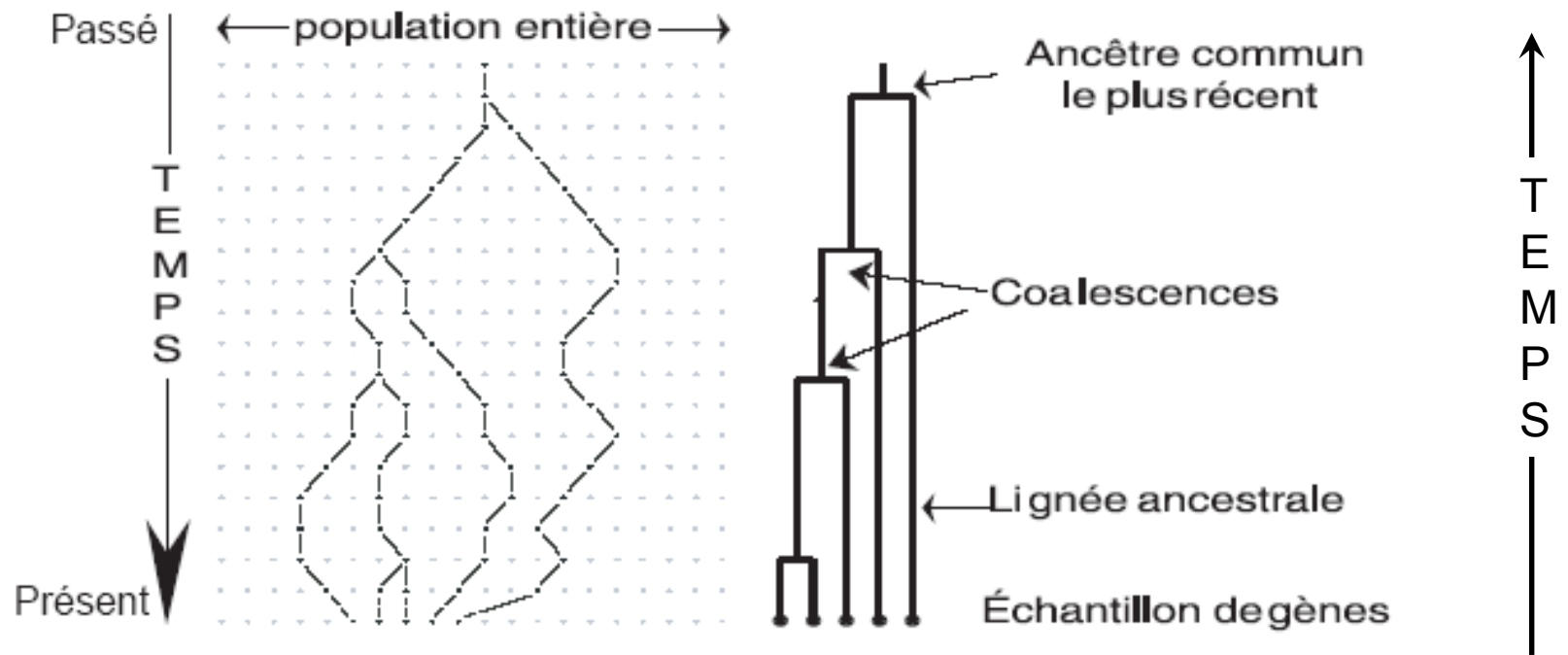


# Plan du cours

1. Principes de la coalescence
  2. Algorithmes de simulations d'arbres et de données
  3. Algorithmes d'estimation de paramètres démographiques fondés sur la coalescence
- Quelques exemples

# Origine de la théorie de la coalescence :

- 1974 –1982 gestation (Kingman, Ewens, Watterson)
- 1982 Kingman & Tajima
- depuis 1990, nombreux développements  
par R. Griffiths, S. Tavaré, R. Hudson, P. Donnelly, J. Felsenstein, R. Nielsen, M. Stephens et beaucoup d'autres...



## → Nouvelle approche de génétique des populations théorique

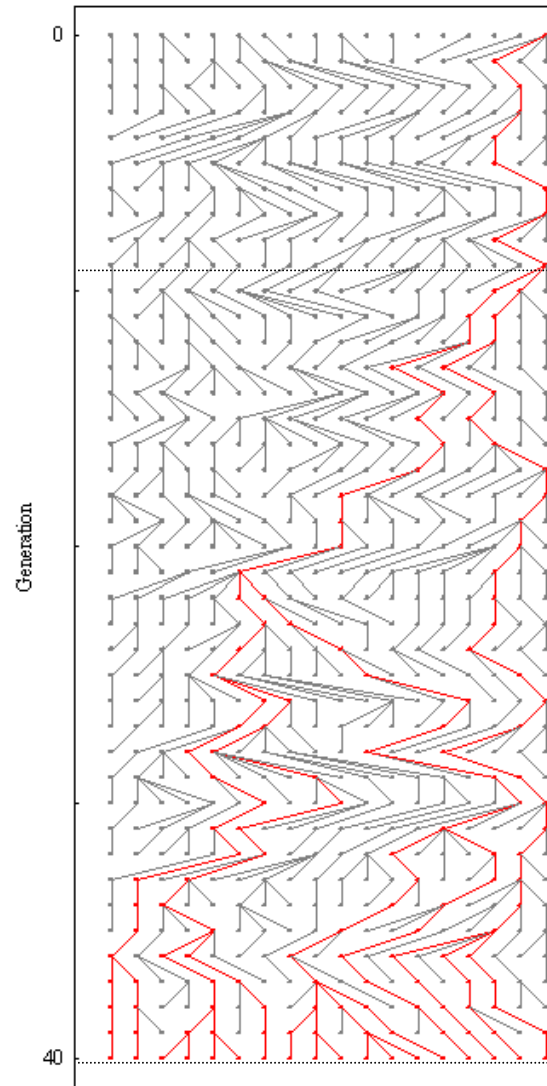
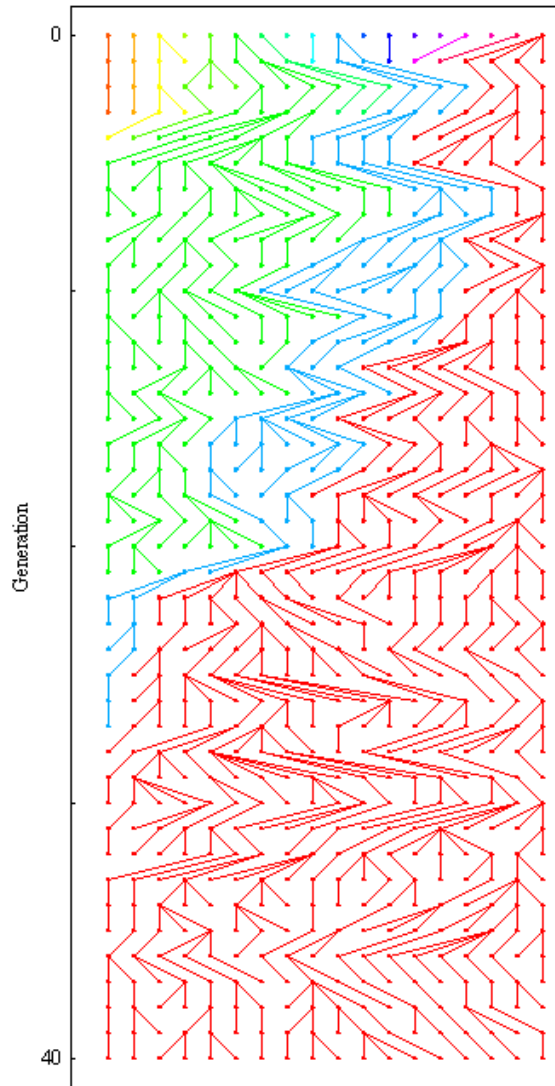
### ☐ Approche classique

- Vision avant (Forward)
- POPULATION
- Fréquences

### ☐ Approche « coalescence »

- Vision arrière (Backward)
- ECHANTILLON
- Généalogie des gènes

# Another way of looking at Genetic drift: *the coalescence theory*



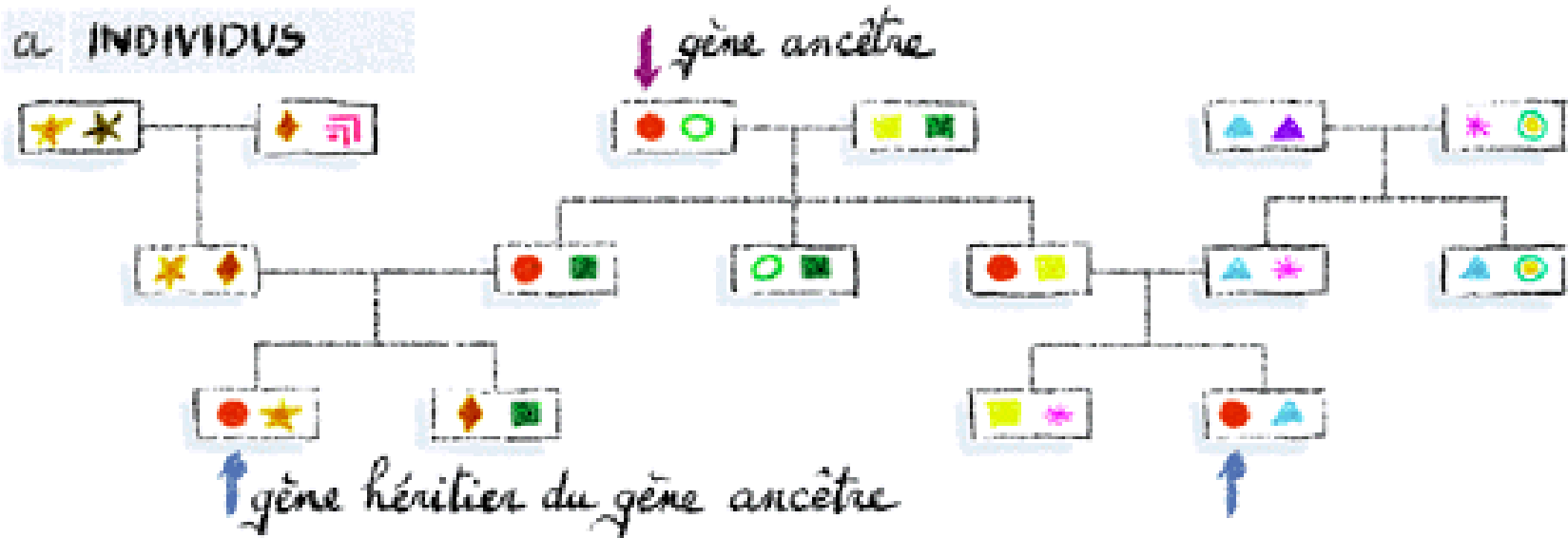
Les différentes  
lignées fusionnent  
(coalescent) au  
fur et à mesure  
que l'on remonte  
vers le passé

Time of coalescence  
( $T$ )

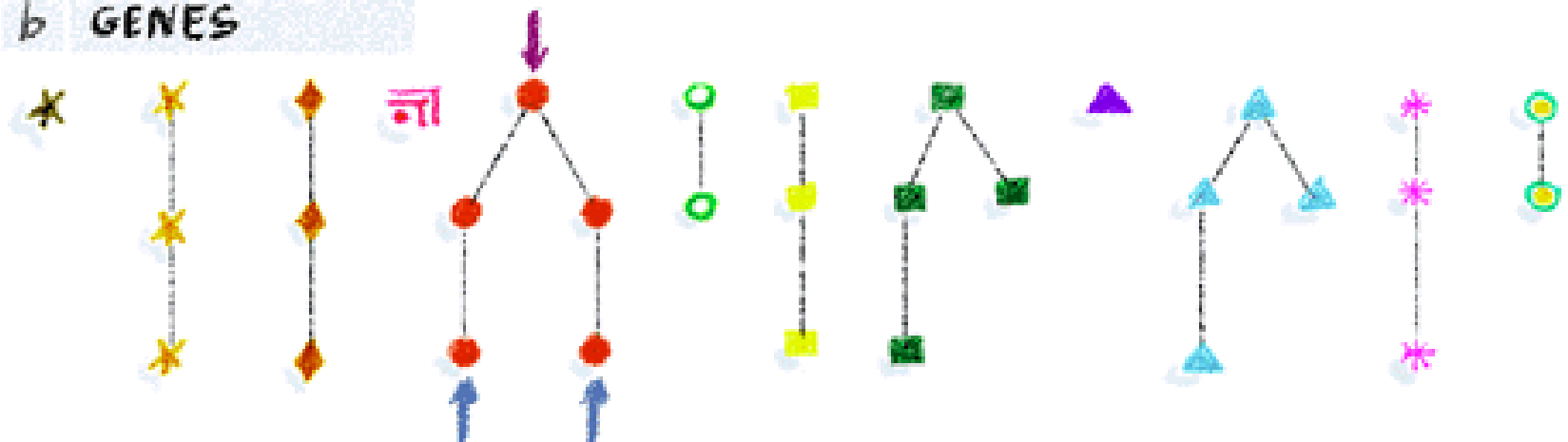
La coalescence : la dérive vue en remontant le temps

# Généalogie des gènes $\neq$ des individus

## a INDIVIDUS

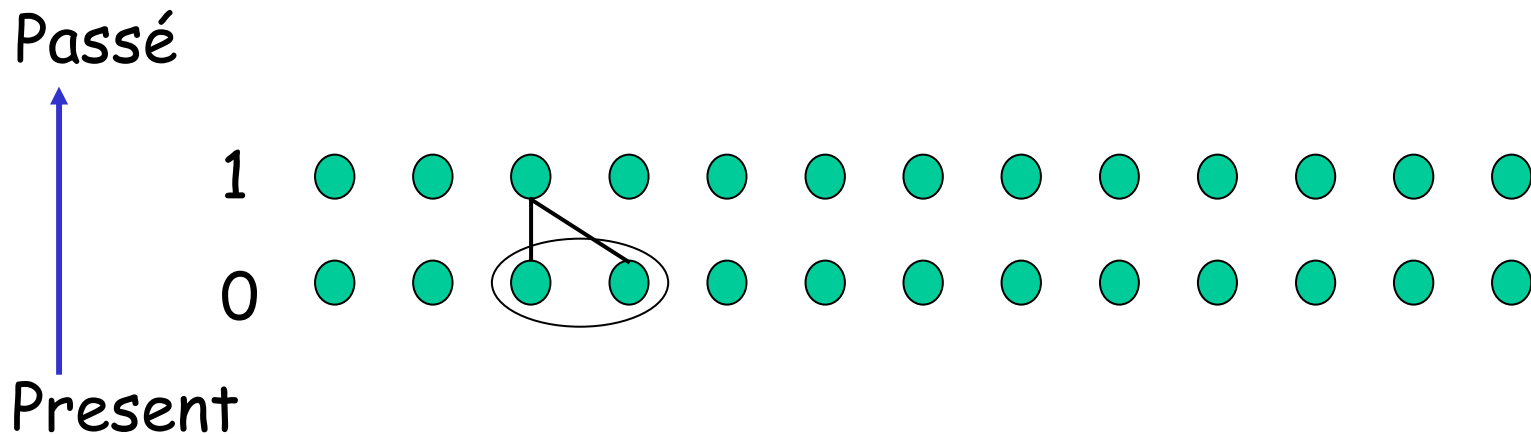


## b GÈNES



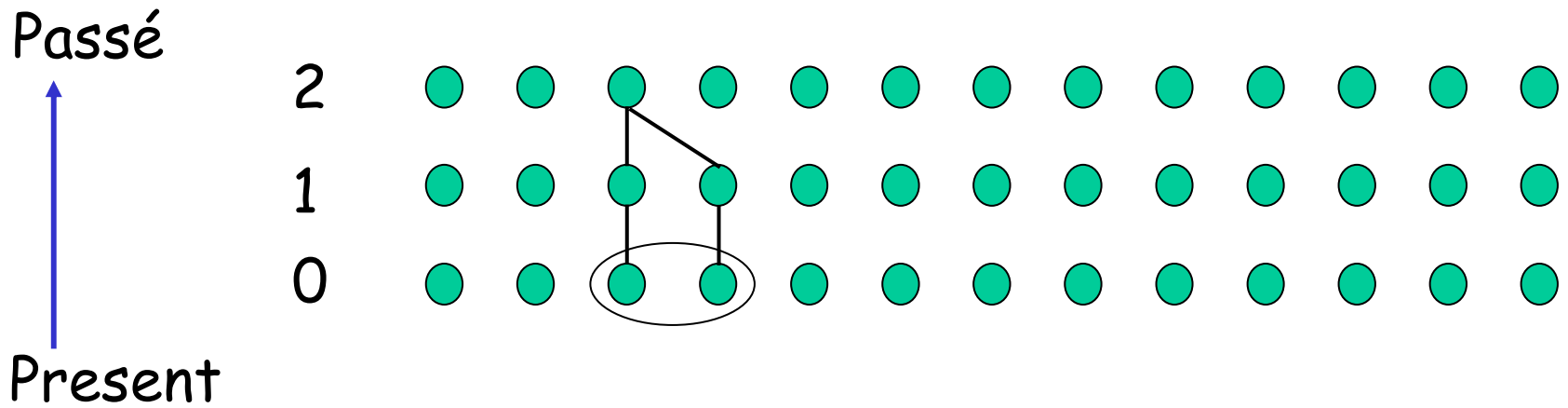
# Probabilité de coalescence en une génération

- Population haploïde de taille  $N$  (*souvent  $2N$  pour diploïdes*).



$$P(T_2 = 1) = \frac{1}{N}$$

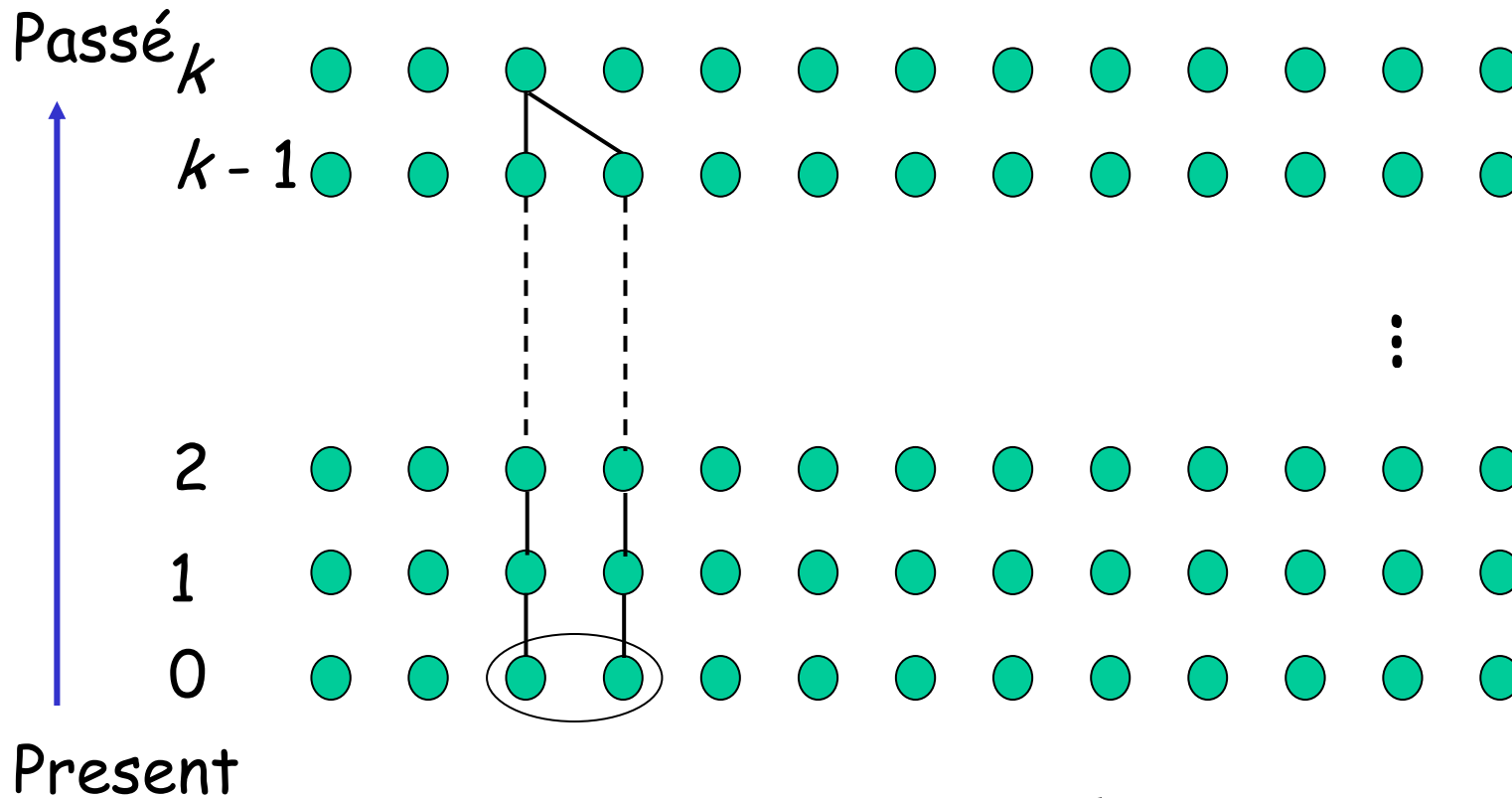
# Probabilité de coalescence en deux générations



$$P(T_2 = 2) = \left(1 - \frac{1}{N}\right) \frac{1}{N}$$



# Probabilité de coalescence en $k$ génération

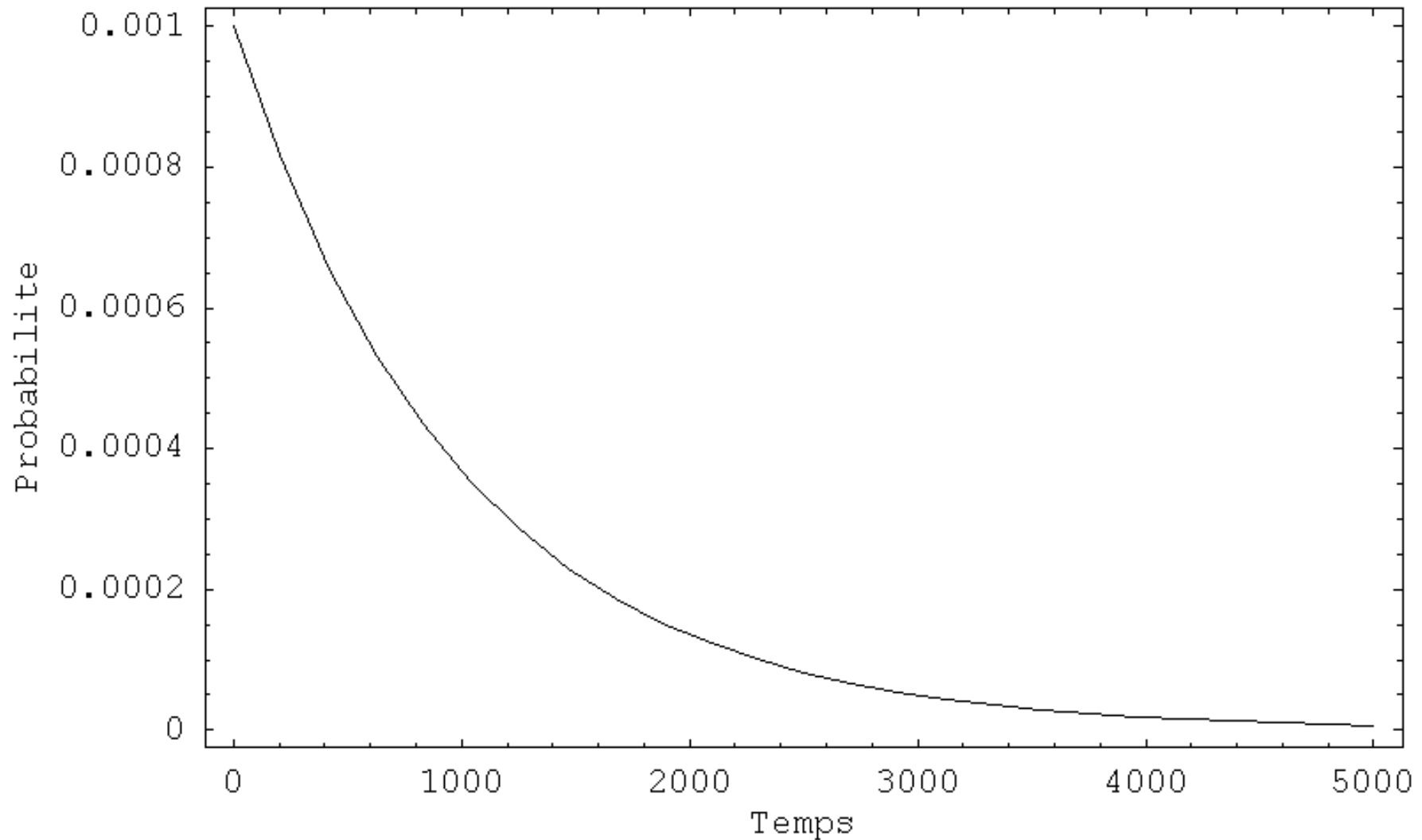


$$P(T_2 = k) = \left(1 - \frac{1}{N}\right)^{k-1} \frac{1}{N}$$

- Loi géométrique de paramètre  $1/N$

# Loi géométrique de paramètre $1/N$

( $N = 1000$ )



Pr(2 lignées coalescent à une génération donnée) =  $\frac{1}{N}$

Pr(2 lignées coalescent k+1 générations en arrière) =

$$\Pr(T_2=k+1) = \left(1 - \frac{1}{N}\right) \dots \left(1 - \frac{1}{N}\right) \left(\frac{1}{N}\right) = \left(1 - \frac{1}{N}\right)^k \left(\frac{1}{N}\right) = \frac{1}{N} e^{k \ln(1 - \frac{1}{N})}$$

Si N suffisamment grand, approximation possible  $\approx \frac{1}{N} e^{-\frac{k}{N}}$

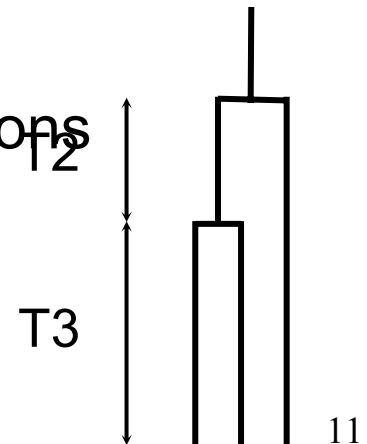
Le temps de coalescence de deux lignées ( longueur des branches) suit une loi de **distribution exponentielle** d'espérance  $N$

[mais classiquement, Unité de temps =  $N$  générations]

$$\square \Pr(T=t) = e^{-t}$$

→ approximation **continue**

d'un processus **discontinu**



## Cas de $j > 2$ lignées ancestrales

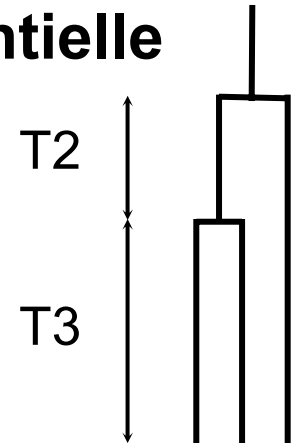
HYPOTHESE : pas de coalescence multiple (OK si  $N$  grand)

$C_j^2 = j(j-1)/2$  paires de lignées peuvent coalescer avec  $\text{Pr} = \frac{1}{N}$

$\text{Pr}(\text{2 lignées parmi } j \text{ coalescent à chaque génération}) = \frac{j(j-1)}{2N}$

**Le temps entre deux coalescences dans un ensemble de  $j$  lignées ancestrales suit une distribution exponentielle d'espérance  $2Ne/(j(j-1))$**

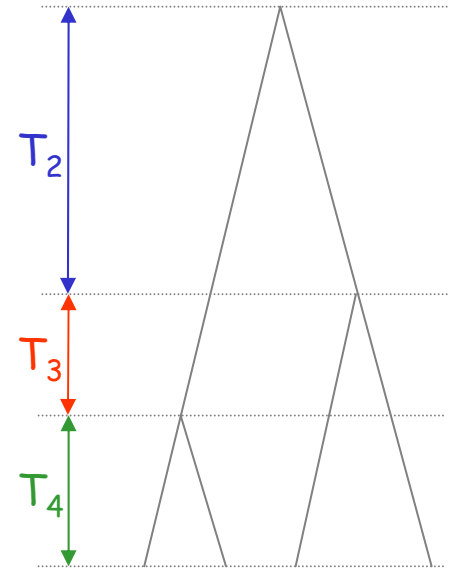
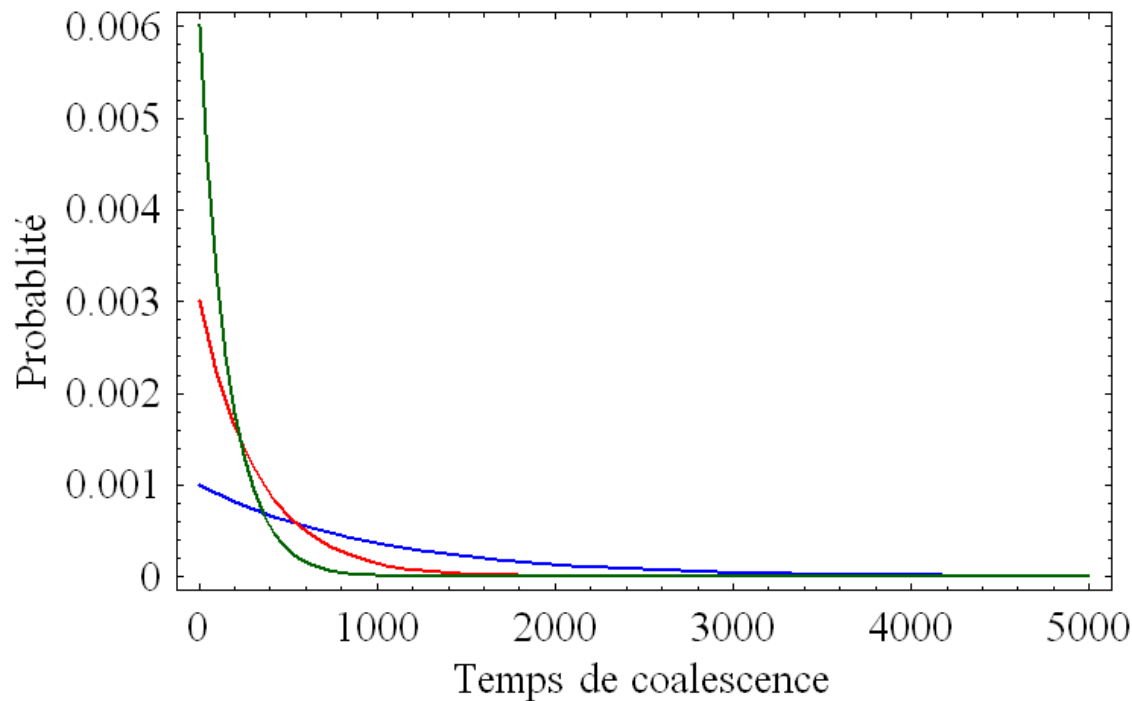
$$\text{Pr}(T_j = k) = \frac{j(j-1)}{2N} e^{-\frac{j(j-1)}{2N} k}$$



Ex: en temps continue (unité =  $N$ ):  $\text{Pr}(T_3=t) = 3 e^{-3t}$

# Temps de coalescence pour des échantillons plus grands

$$p(T_j = k) = \frac{j(j-1)}{2N} \left(1 - \frac{j(j-1)}{2N}\right)^{k-1}$$



$$E(T_j) = \frac{2N}{j(j-1)}$$

$$\text{var}(T_j) = \frac{4N^2}{j^2(j-1)^2}$$

# Principaux avantages de la coalescence

- la généalogie sous-jacente, et plus généralement l'histoire évolutive d'un échantillon, est le grand inconnu en évolution et ne peut pas être "refait" □ la coalescence permet de bien prendre en compte cette inconnue
- Simplification de l'analyse quantitative des modèles stochastiques et réinterprétation des résultats théoriques
- La structure des données génétiques reflète pour une large part la généalogie sous-jacente □ facilite l'analyse de la variabilité génétique observée et la compréhension des phénomènes évolutifs ayant agi dessus

# Principaux avantages de la coalescence

- **Méthodes de simulation extrêmement efficaces**
- **Elle fournit de puissantes techniques d'inférences de paramètres évolutifs (démographiques, génétiques,...) à partir des données génétiques, dont certaines permettent l'usage complet de l'information contenue dans les données (Maximum de vraisemblance, approches Bayésiennes)**

# Construction d' Arbres de coalescence

- Principe général :

- Marqueurs neutres = nombre de descendants indépendant du type allélique
- > processus démographiques découplés des processus mutationnels
- Construction en 2 temps :
  - (1) on construit l'arbre :  
topologie + longueurs de branches
  - (2) on ajoute les mutations



# Simulation d'arbres de coalescence

(cf : SimulationArbres.pdf)

## ➤ Méthode 1 : modèle en urne

très rapide mais ne marche que dans le cas d'une population panmictique sans fluctuations démographiques

## ➤ Méthode 2 : approximations continues de Hudson

Assez rapide mais ne marche pas dans des cas complexes (petites pops, forts taux de migration, modèles très complexes)

## ➤ Méthode 3 : génération par génération

Ok pour tous modèles, mais **lent**

**RAPIDITE :**

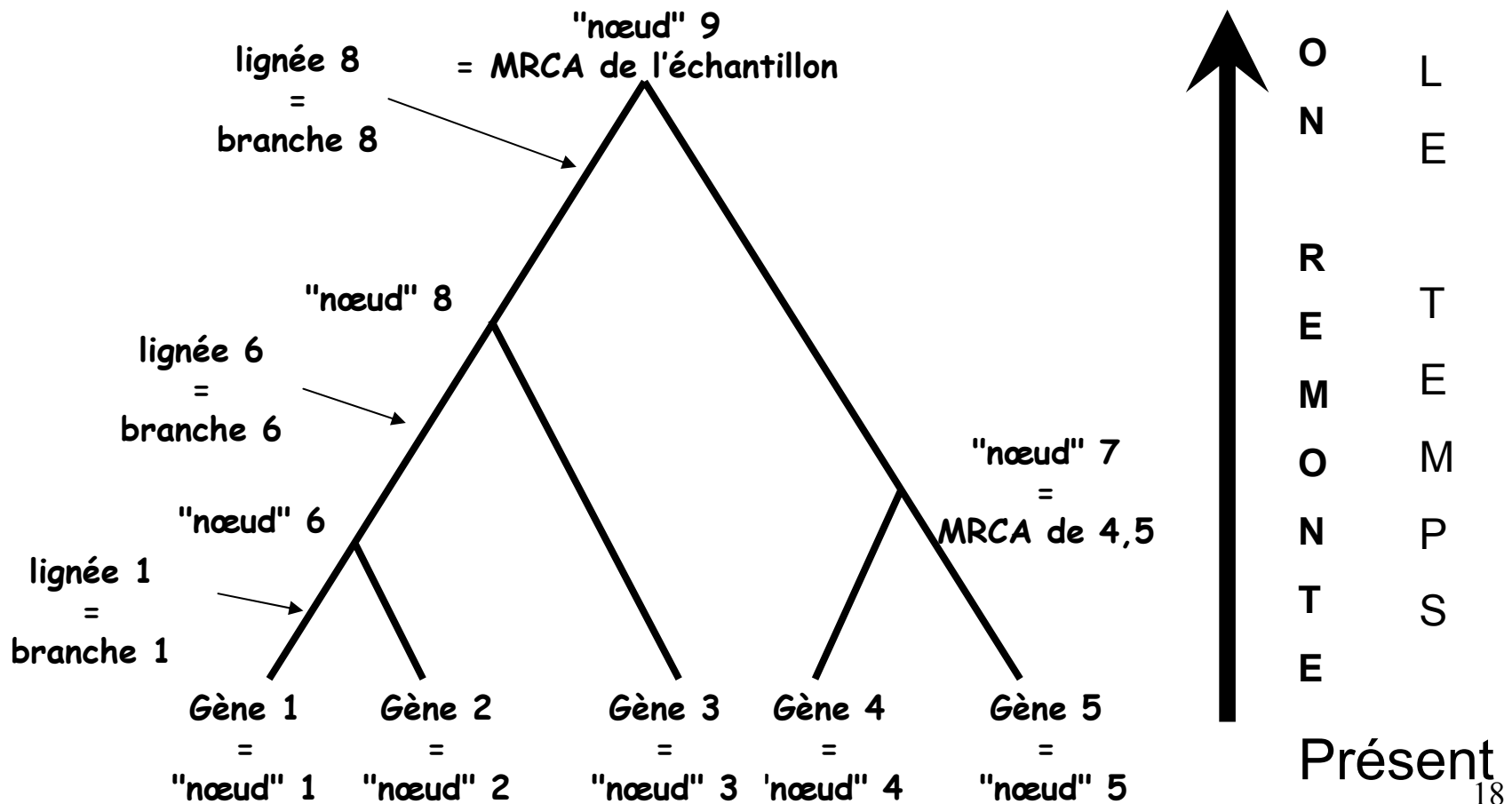
*Urne > Approximation continue > Génération par génération*

**FLEXIBILITE :**

*Génération par génération > Approximation continue > Urne*

# Construction d' Arbres de coalescence

- Représentation de l'arbre :



# Construction d' Arbres de coalescence:

## (1) Génération par génération

- Principe simple, sans approximations :
  - On remonte dans le passé génération par génération
  - À chaque génération, on recherche les éventuels événements affectant la généalogie (coalescence,  $\pm$  migration)
  - On s'arrete quand on arrive a l'ancêtre commun de tous les gènes de l'échantillon  
= MRCA (Most Recent Common Ancestor)

# Construction d' Arbres de coalescence:

## (1) Génération par génération

- Exemple :
  - échantillon de 4 gènes
  - à un locus neutre
  - ayant évolué dans une population haploïde panmictiques de taille  $N=10$

# Construction d'Arbres de coalescence:

## (1) Génération par génération

- **Exemple :** éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	1	2	3	4
Nombre aléatoire entre 1 et N assigné aux nœuds				
Génération d'apparition du nœud/lignée	0	0	0	0

$G_n=0$

① ② ③ ④

# Construction d'Arbres de coalescence:

## (1) Génération par génération

- Exemple : éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	1	2	3	4
Nombre aléatoire entre 1 et N assigné aux nœuds				
Génération d'apparition du nœud/lignée	0	0	0	0

Probabilité d'avoir  
une coalescence  
parmi  $j$  lignées à  
une génération

$$= j(j-1)/2N$$

= probabilité de tirer  
2 nombre identiques  
entre 1 et  $N$  parmi  $j$   
tirages

$G_n=0$

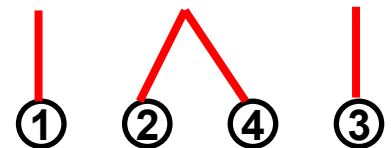
# Construction d'Arbres de coalescence:

## (1) Génération par génération

- Exemple : éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	1	2	3	4
Nombre aléatoire entre 1 et N assigné aux nœuds	2	6	5	6
Génération d'apparition du nœud/lignée	0	0	0	0

Coalescence à la génération 1 des nœuds/lignées 3 et 4



$G_n=1$

# Construction d'Arbres de coalescence:

## (1) Génération par génération

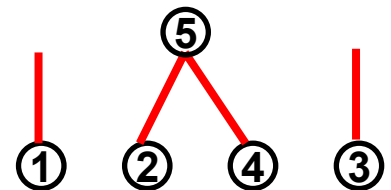
- Exemple : éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	1	3	5
Nombre aléatoire entre 1 et N assigné aux nœuds	2	5	6
Génération d'apparition du nœud/lignée	0	0	1

$G_n=1$

Coalescence à la génération 1 des nœuds/lignées 3 et 4

Donne le nœud 5





# Construction d'Arbres de coalescence:

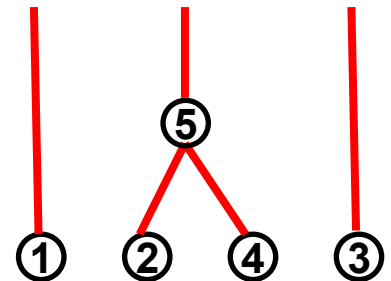
## (1) Génération par génération

- **Exemple :** éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	1	3	5
Nombre aléatoire entre 1 et N assigné aux nœuds	3	1	7
Génération d'apparition du nœud/lignée	0	0	1

$G_n=2$

Rien à la génération 2



# Construction d'Arbres de coalescence:

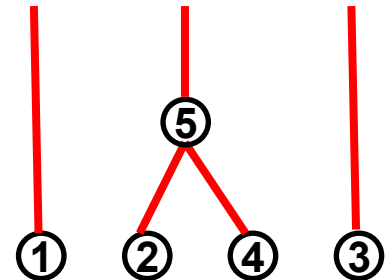
## (1) Génération par génération

- **Exemple :** éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	1	3	5
Nombre aléatoire entre 1 et N assigné aux nœuds	7	4	8
Génération d'apparition du nœud/lignée	0	0	1

$G_n=3$

Rien à la génération 3



# Construction d'Arbres de coalescence:

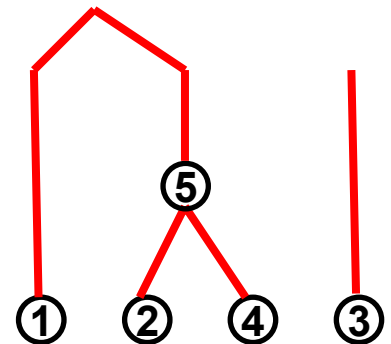
## (1) Génération par génération

- Exemple : éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	1	3	5
Nombre aléatoire entre 1 et N assigné aux nœuds	5	2	5
Génération d'apparition du nœud/lignée	0	0	1

$G_n=4$

Coalescence à la génération 4 des nœuds/lignées 1 et 5



# Construction d'Arbres de coalescence:

## (1) Génération par génération

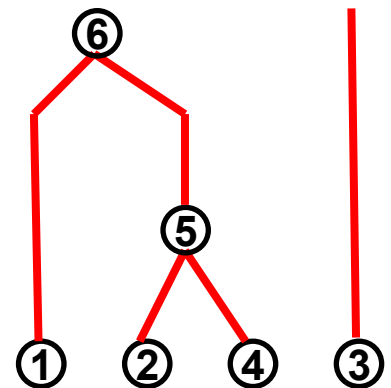
- Exemple : éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	3	6
Nombre aléatoire entre 1 et N assigné aux nœuds	2	5
Génération d'apparition du nœud/lignée	0	5

$G_n=4$

Coalescence à la génération 4 des nœuds/lignées 1 et 5

Donne le noeuds 6



# Construction d'Arbres de coalescence:

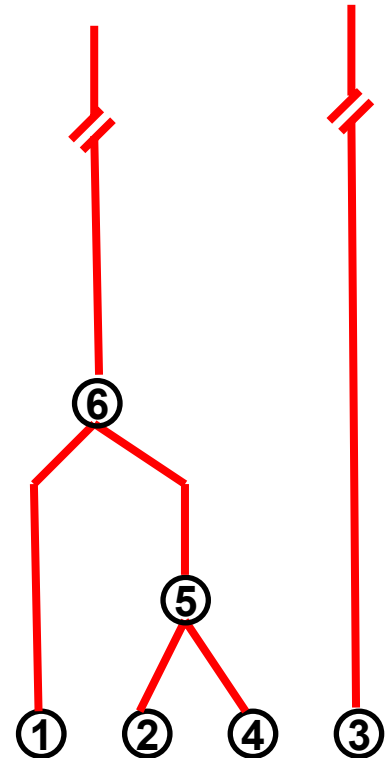
## (1) Génération par génération

- Exemple : éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	3	6
Nombre aléatoire entre 1 et $N$ assigné aux nœuds	3	9
Génération d'apparition du nœud/lignée	0	5

$G_n=5$

Rien aux générations 5,6,...



# Construction d'Arbres de coalescence:

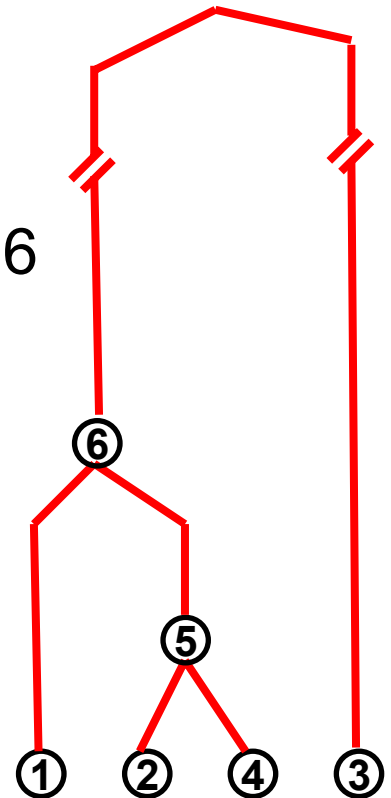
## (1) Génération par génération

- Exemple : éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	3	6
Nombre aléatoire entre 1 et $N$ assigné aux nœuds	7	7
Génération d'apparition du nœud/lignée	0	5

$G_n=20$

Coalescence à la génération 20 des 2 dernières lignées 3 et 6



# Construction d'Arbres de coalescence:

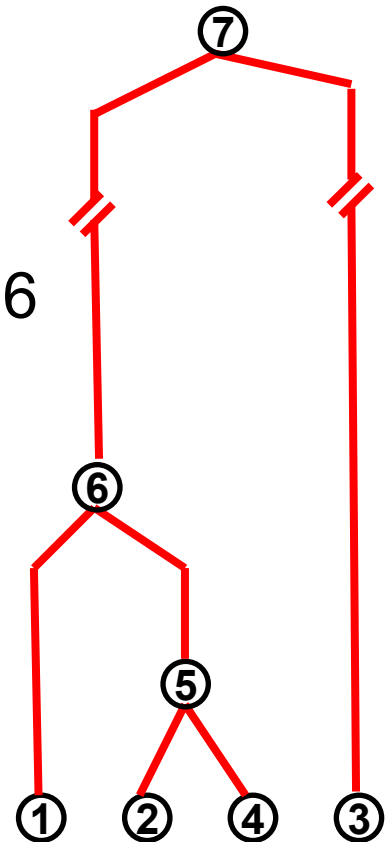
## (1) Génération par génération

- Exemple : éch. 4 gènes, une pop.  $N=10$

Numéro des nœuds/lignées	3	6
Nombre aléatoire entre 1 et $N$ assigné aux nœuds	7	7
Génération d'apparition du nœud/lignée	0	5

Coalescence à la génération 20 des 2 dernières lignées 3 et 6

Donne le nœuds 7 = MRCA de l'échantillon



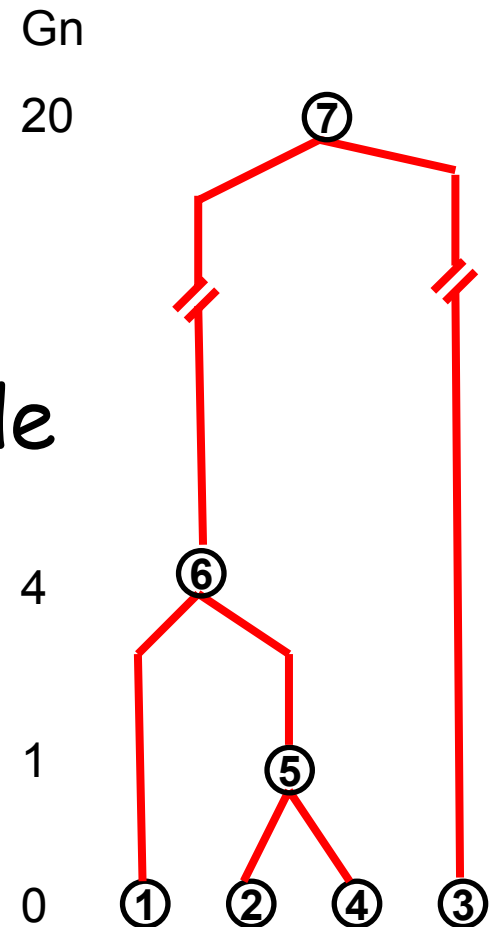
$G_n=20$

# Construction d'Arbres de coalescence: (1) Génération par génération

- Exemple :  
éch. 4 gènes, une pop.  $N=10$

L'arbre, topologie et longueurs de branches, est construit

Mutations ajoutées plus tard...





# Construction d' Arbres de coalescence:

## (2) Approximation continues de Hudson

- Principe : 2 étapes successives
  - (1) construire la topologie en coalesçant au hasard les lignées ancestrales
  - (2) simuler les temps entre 2 coalescences successives = longueurs des branches

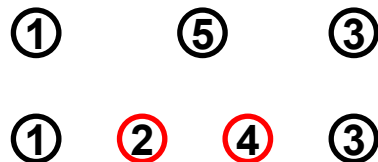
# Construction d' Arbres de coalescence:

## (2) Approximation continues de Hudson

- Exemple : éch. 4 gènes, une pop  $N=10$

(1) construire la topologie en coalesçant au hasard les lignées ancestrales

1ere coalescence = tirage au sort de 2 lignées parmi 4 → les lignées 2 et 4 donnent la lignée 5



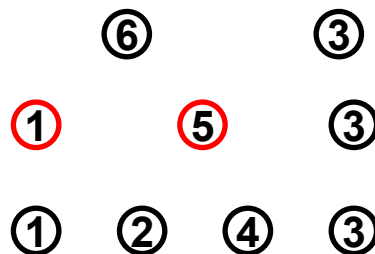
# Construction d' Arbres de coalescence:

## (2) Approximation continues de Hudson

- Exemple : éch. 4 gènes, une pop  $N=10$

(1) construire la topologie en coalesçant au hasard les lignées ancestrales

2eme coalescence = tirage au sort de 2 lignées parmi les 3 restantes  $\rightarrow$  les lignées 1 et 5 donnent la lignée 6



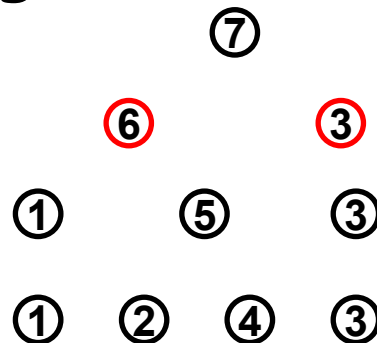
# Construction d' Arbres de coalescence:

## (2) Approximation continues de Hudson

- Exemple : éch. 4 gènes, une pop  $N=10$

(1) construire la topologie en coalesçant au hasard les lignées ancestrales

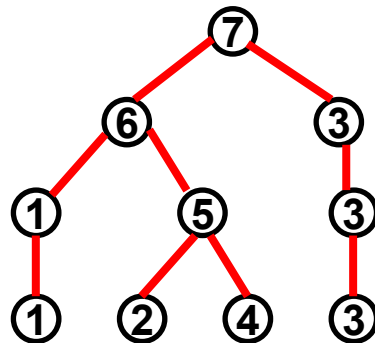
3eme coalescence = seules les 2 dernières peuvent coalescer  $\rightarrow$  les lignées 6 et 3 donnent la lignée 7



# Construction d' Arbres de coalescence:

## (2) Approximation continues de Hudson

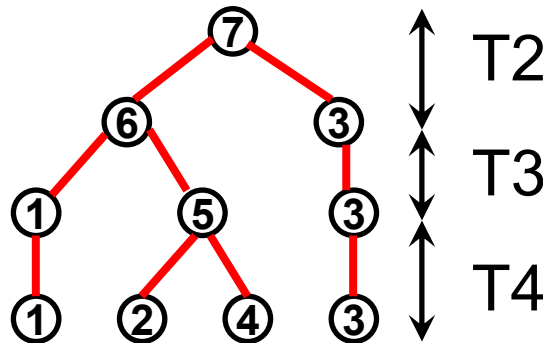
- Exemple : éch. 4 gènes, une pop  $N=10$ 
  - (1) construire la topologie en coalesçant au hasard les lignées ancestrales



# Construction d' Arbres de coalescence:

## (2) Approximation continues de Hudson

- Exemple : éch. 4 gènes, une pop  $N=10$
- (2) simuler les temps entre 2 coalescences successives = longueurs des branches
- 3 longueurs de branches a simuler  $T_4, T_3, T_2$



# Construction d'Arbres de coalescence:

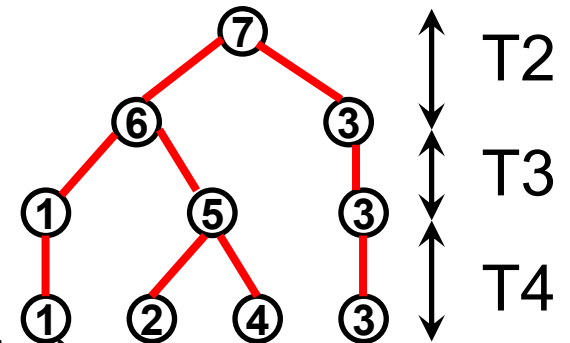
## (2) Approximation continues de Hudson

- Exemple : éch. 4 gènes, une pop  $N=10$
- (2) simuler les temps entre 2 coalescences successives = longueurs des branches
- 3 longueurs de branches a simuler  $T_4, T_3, T_2$

$$\Pr(T_j = k) = \frac{j(j-1)}{2N} e^{-\frac{j(j-1)}{2N}k}$$

$T_4$  tiré dans une loi exponentielle de paramètre

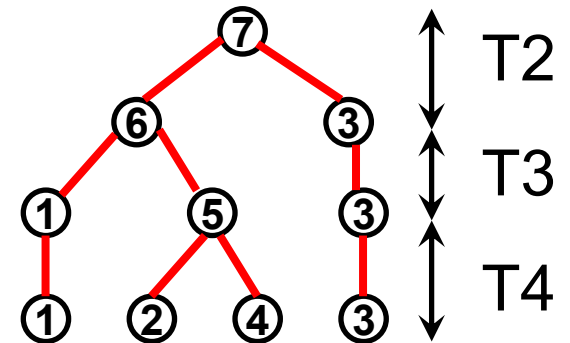
$$2N/j(j-1) = 2 \cdot 10 / 4 \cdot 3 \text{ (algorithmes disponibles)}$$



# Construction d' Arbres de coalescence:

## (2) Approximation continues de Hudson

- Exemple : éch. 4 gènes, une pop  $N=10$
- (2) simuler les temps entre 2 coalescences successives = longueurs des branches
- 3 longueurs de branches a simuler  $T_4, T_3, T_2$
- $T_4$  tiré dans une loi exponentielle de paramètre  $2N/j(j-1)=2*10/4*3 \rightarrow 1,2$
- $T_3$  dans  $\exp(2*10/3*2) \rightarrow 2,6$
- $T_2$  dans  $\exp(2*10/2*1) \rightarrow 15,7$

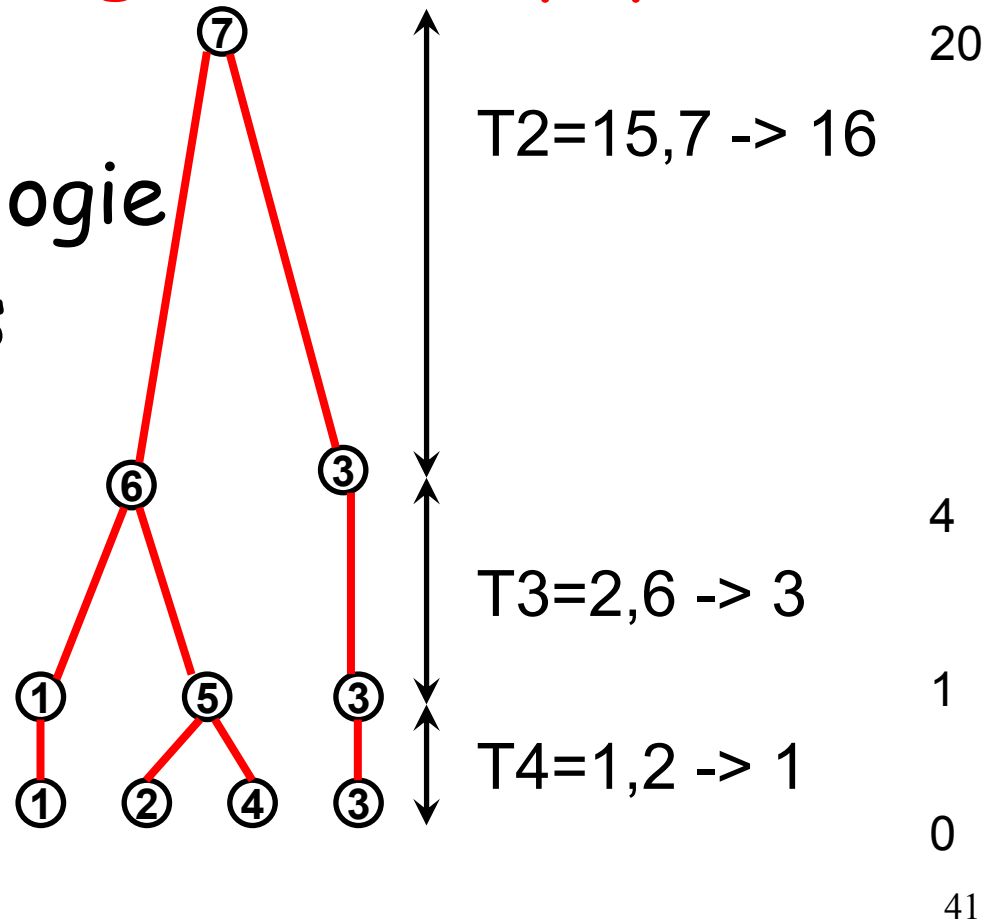




# Construction d'Arbres de coalescence: (2) Approximation continues de Hudson

- Exemple : éch. 4 gènes, une pop  $N=10$   $G_n$

On a donc la topologie  
Et la longueur des  
branches



# Construction d' Arbres de coalescence:

## (3) Ajout des mutations sur l'arbre

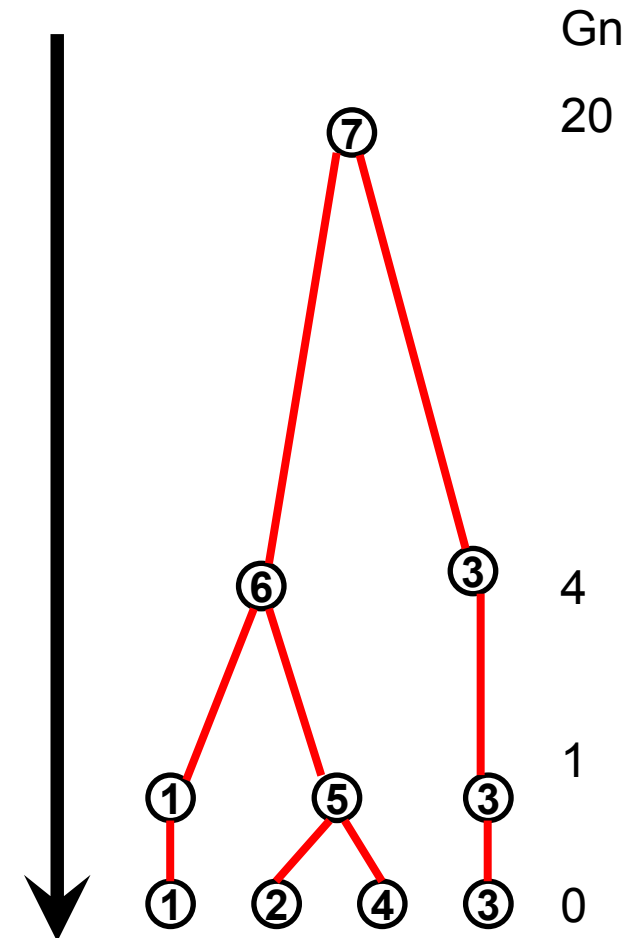
Principe général :

On distribue les mutations sur les différentes branches de l'arbre en descendant du haut vers le bas en fonction du taux de mutation  $\mu$

Chaque mutation induit un changement de l'état allélique du nœud descendant

Selon le modèle mutationnel choisi (= reflète le processus mutationnel) :

IAM, KAM, SMM, GSM, ...



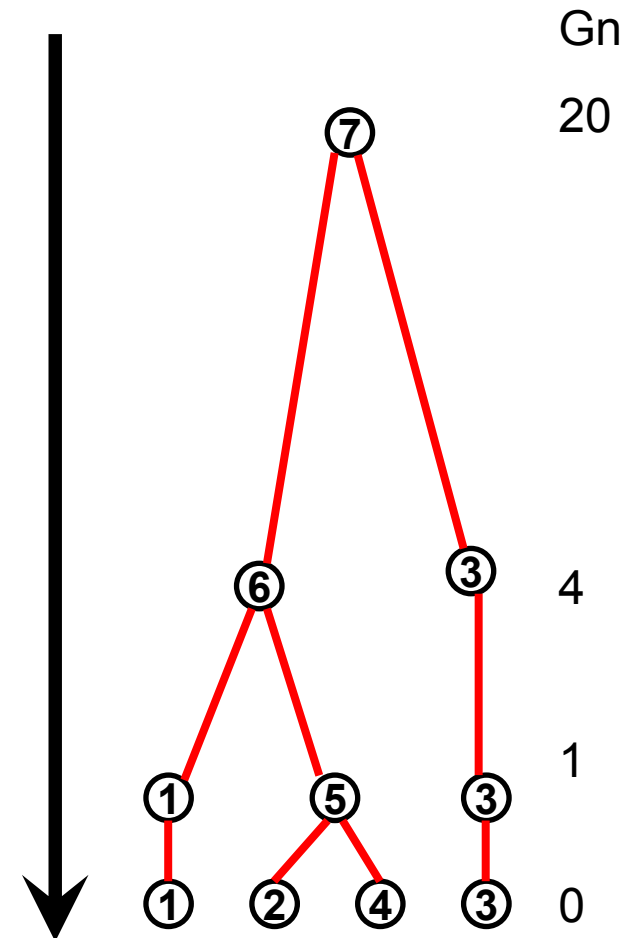
# Construction d' Arbres de coalescence:

## (3) Ajout des mutations sur l'arbre

Sur une branche de longueur  $t$ , le nombre de mutation suit une loi binomiale de paramètres  $(\mu, t)$

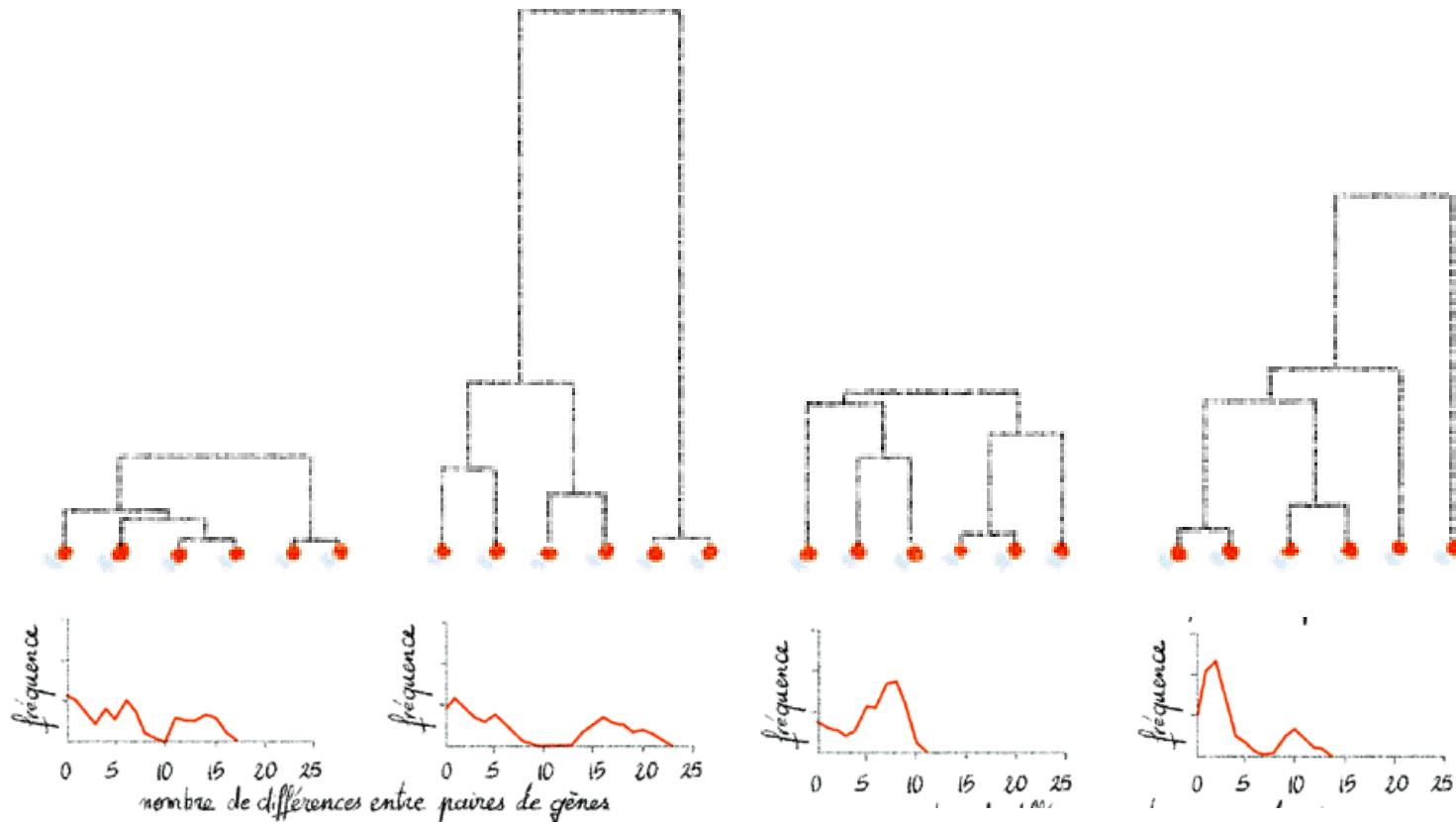
Approximation loi de poisson de paramètre  $(\mu^* t)$

$$\Pr(\text{nb de mut} = k) = \frac{\mu t^k e^{-\mu t}}{k!}$$



# A quoi servent ces arbres de coalescence (généalogies)?

- Étudier l'effet de certains paramètres sur la forme de l'arbre et sur le polymorphisme d'un échantillon



# A quoi servent ces arbres de coalescence (généalogies)?

- Étudier l'effet de certains paramètres sur la forme de l'arbre, sur le polymorphisme d'un échantillon et sur des statistiques résumées.
- Créer des échantillons simulés, par exemple pour tester des méthodes d'estimation
- Estimer des paramètres évolutifs avec certaines méthodes spécifiques (Approximate Bayesian Computations)

# coalescence et estimation de paramètres évolutifs

- Par maximum de vraisemblance (par extension, ce fait aussi dans un cadre bayésien)

Utilise des algorithmes spécifiques, fondés sur la coalescence, pour calculer (ou estimer) la vraisemblance d'un échantillon

- ☐ **utilise toute l'information des données**

- Approximate Bayesian Computation (ABC)

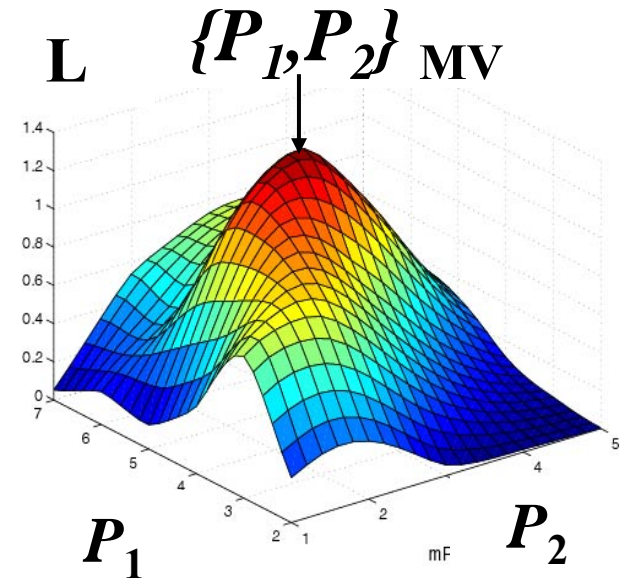
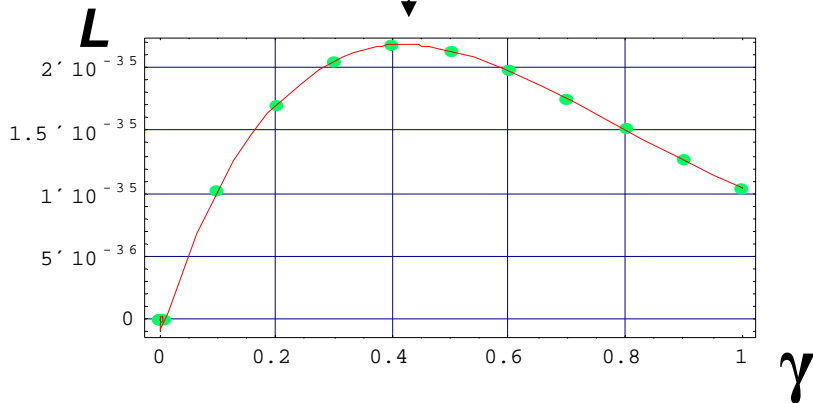
Utilise les algorithmes de simulation de données décrit précédemment et des statistiques résumées (ex:  $n_A$ ,  $H_e$ ,  $F_{st}$ ,  $VarAll...$ ) pour approcher la vraisemblance d'un échantillon par algorithme d'acceptation/rejet

- ☐ **résume l'information des données en plusieurs statistiques**

# Principe de l'estimation par maximum de vraisemblance

On calcule/estime la vraisemblance  $[L = \Pr(\text{échant.} | \text{paramètres})]$  de l'échantillon pour chaque valeur du paramètre

→ Courbe/surface de vraisemblance  
 $\gamma_{MV}$



Estimateur MV=valeur qui maximise cette courbe/surface

→ **Pb 1 : bcp de paramètres → grand espace des paramètres à explorer**

# Maximum de vraisemblance et coalescence

Notations :  $P$ =ensemble des paramètres;  $D$ =données= échantillon;  
 $G$ =ensemble des généalogies possibles;  $G_k$ =une généalogie;  $E$ =espérance

- Pas de formule explicite pour la vraisemblance  $L(P|D)=\Pr(D|P)$ , par contre on sait calculer la probabilité des données sachant les paramètres et la généalogie  $\Pr(D|G;P)$ , on utilise alors :

$$L(P|D) = \int_G \Pr(D|G;P) \Pr(G;P)$$

Somme sur toutes les généalogies possibles

□ impossible à faire !!!

- Simulation de Monte Carlo : on prend la moyenne sur un grand nombre  $K$  de généalogies  $G_k$  simulées:

$$L(P|D) = E[\Pr(D|G;P)] \approx \frac{1}{K} \sum_{k=1}^K \Pr(D|G_k;P)$$

→ **Pb 2 : Beaucoup de généalogies à simuler pour avoir une bonne estimation de la vraisemblance**



# Maximum de vraisemblance et coalescence

2 grand espace a explorer : paramètres et généalogies

→ Pb 1 : plus de paramètres → plus grand espace des paramètres à explorer

→ Pb 2 : Beaucoup de généalogies à simuler pour avoir une bonne estimation de la vraisemblance

→ Pb 3 : plus de paramètres → généalogies plus complexes (plus d'événements possibles) → plus grand espace des généalogies

Donc plus le modèle a de paramètres plus il faut de temps (ou des algorithmes plus efficaces) pour explorer l'espace des paramètres mais aussi celui des généalogies.

→ Souvent très long temps de calculs

# **ML OK pour une population panmictique isolée (car peu de paramètres) :**

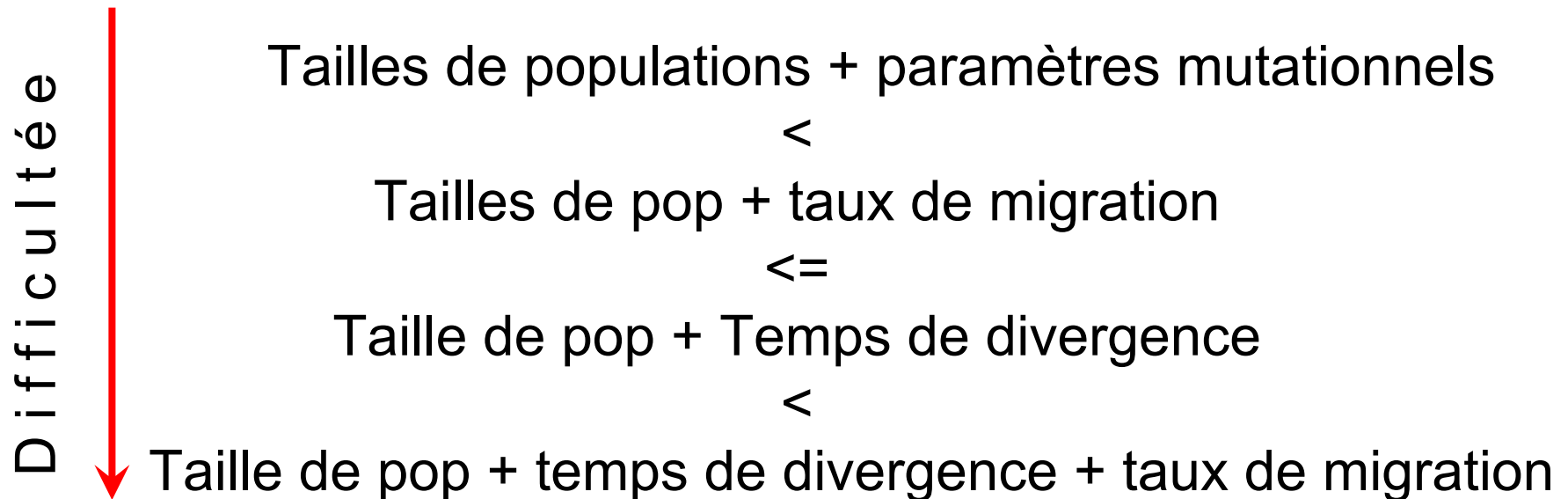
- Algorithmes très efficaces (ex : IS stephens & donnelly 2001)
- Utilisés entre autre pour :
  - Détection de goulet d'étranglement ([Bottleneck](#))
  - Détection de migrants par méthodes d'assignation([Genclass](#))
  - Estimation de paramètres mutationnels (microsats)
  - Estimation de taille efficaces (avec echantillonnage temporel)

**Plus problématique pour des cas de populations structurées...  
(car beaucoup de paramètres et migration complexe à prendre en compte)**

# Source des problèmes d'estimation par MV

Maximum de vraisemblance OK pour une population panmictique isolée mais plus problématique pour des cas de populations structurées...

Les problèmes d'estimation ne viennent pas seulement du nombre de paramètres à estimer mais aussi de quels paramètres on veut estimer :



# **Estimation de paramètres démographiques (taux de migration) par maximum de vraisemblance**

- **Présentation des deux principales classes d'algorithmes existants**
- **Différences, avantages et inconvénients des deux méthodes**
- **Robustesse: quelques résultats préliminaires (thèse, d'autres exemples demain)**

# Calcul / estimation de la vraisemblance :

➤ L'approche de **Griffiths et coll. (GENETREE)** utilisant des chaînes de Markov absorbantes et de l'*Importance Sampling (IS)*

➤ L'approche de **Felsenstein et coll. (MIGRATE)** utilisant un algorithme de *Monte Carlo par Chaînes de Markov (MCMC)*

**Différences = exploration des espaces de paramètres et des généalogies**

**MCMC explore en même temps les paramètres et les généalogies avec algo de Metropolis-Hasting**

**IS explore que les généalogies, paramètres explorer indépendamment (Latin hypercube sampling on parameter range)**

# Chaînes de Markov absorbantes et *Importance Sampling (IS)*

Notation :  $n_{t0}=Data=sample$   $\square$   $L(P|D) = \Pr(n_{t0}|P)$

- La récurrence de base

$$\Pr(n_t|P) = f(n_t) \cdot \sum_{n_{t'}} [IS(n_t \rightarrow n_{t'}|P) \cdot \Pr(n_{t'}|P)]$$

Diagram illustrating the recurrence formula with annotations:

- $\Pr(n_t|P)$ : État de l'échantillon au moment  $t$
- $f(n_t)$ : État de l'échantillon au moment  $t$
- $IS(n_t \rightarrow n_{t'}|P)$ : Transition entre 2 états ancestraux : Coalescence, Mutation ou migration
- $\Pr(n_{t'}|P)$ : État de l'échantillon au moment  $t'$  ( $=t+1$  événement)

On crée un arbre de coalescence possible en remontant le temps événement par événement (= à chaque fois que l'échantillon change de configuration) jusqu'au MRCA

Ce sont les chaînes de Markov absorbantes (MRCA = état absorbant) qui explore l'espace des généalogies

# Griffiths et coll. : Chaînes de Markov absorbantes et de l'Importance Sampling (IS)

Notation :  $n_{t_0} = \text{Data} = \text{sample}$   $\square$   $L(P|D) = \Pr(n_{t_0}|P)$

- La récurrence de base

$$\Pr(n_t|P) = f(n_t) \cdot \sum_{n_{t'}} [IS(n_t \rightarrow n_{t'}|P) \cdot \Pr(n_{t'}|P)]$$

Diagram annotations:

- Red circle around  $n_t$  in  $\Pr(n_t|P)$  with an arrow pointing to "État de l'échantillon au moment  $t$ ".
- Red circle around  $n_{t'}$  in  $\Pr(n_{t'}|P)$  with an arrow pointing to "État de l'échantillon au moment  $t'$  ( $=t+1$  événement)".
- Red circle around the  $IS(n_t \rightarrow n_{t'}|P)$  term with an arrow pointing to "Transition entre 2 états ancestraux : Coalescence, Mutation ou migration".

La fonction d'IS permet de trouver les événements les plus probables et donc d'explorer des zones de fortes probabilités dans l'espace des généalogies (compensation par les poids  $f(n_t)$  de l'IS)

État de l'échantillon au moment  $t$

Transition entre 2 états ancestraux :  
 Coalescence  
 Mutation  
 migration

État de l'échantillon au moment  $t'$  ( $=t+1$  événement)

$$\Pr(n_t | P) = f(n_t) \cdot \sum_{n_{t'}} [IS(n_t \rightarrow n_{t'} | P) \cdot \Pr(n_{t'} | P)]$$

$$p(\mathbf{n}(t) = \eta) = \frac{1}{\left( \sum_a \frac{n_a(n_a-1)}{4N} + \sum_a n_a \mu + \sum_a \sum_{b, b \neq a} n_a m_{ab} \right)}$$

$$\times \left[ \sum_a \left( n_a \mu \sum_i \sum_{j: n_{aj} > 0, j \neq i} \frac{n_{ai} + 1}{n_a} p_{ij} p(\eta - \mathbf{e}_{aj} + \mathbf{e}_{ai}) \right) \text{ Mut} \right.$$

$a = \text{pops}$

$$+ \sum_a \sum_{b, b \neq a} \left( n_b m_{ab} \sum_{i: n_{bi} > 0} \frac{n_{ai} + 1}{n_a + 1} p(\eta - \mathbf{e}_{bi} + \mathbf{e}_{ai}) \right) \text{ Mig}$$

$i, j = \text{états alléliques}$

$$+ \sum_a \left( \frac{n_a(n_a-1)}{4N} \sum_{j: n_{aj} > 1} \frac{n_{aj} - 1}{n_a - 1} p(\eta - \mathbf{e}_{aj}) \right) \Bigg]. \text{ Coa}$$

$m = \text{migration}$



État de l'échantillon au moment t

Transition entre 2 états ancestraux :  
Coalescence  
Mutation  
migration

État de l'échantillon au moment t' (=t+1 événement)

$$\Pr(n_t|P) = f(n_t) \cdot \sum_n [IS(n_t \rightarrow n_{t'}|P) \cdot \Pr(n_{t'}|P)]$$

L'équation (4.34) peut être simplifiée en considérant  $\theta = 4N\mu$ ,  $\gamma_{ab} = 4Nm_{ab}$ ,  $\gamma_a = \sum_{b,b \neq a} \gamma_{ab}$  et  $\beta = \sum_a n_a(n_a - 1 + \gamma_a + \theta)$ , on obtient alors

$$\begin{aligned}
p(\mathbf{n} = \eta) = & \frac{1}{\beta} \sum_a \left[ \theta \sum_i \sum_{j: n_{aj} > 0, j \neq i} (n_{ai} + 1) p_{ij} p(\eta - \mathbf{e}_{aj} + \mathbf{e}_{ai}) \right. \\
& + \sum_{b, b \neq a} n_b \gamma_{ab} \sum_{i: n_{bi} > 0} \frac{n_{ai} + 1}{n_a + 1} p(\eta - \mathbf{e}_{bi} + \mathbf{e}_{ai}) \\
& \left. + n_a \sum_{j: n_{aj} > 1} (n_{aj} - 1) p(\eta - \mathbf{e}_{aj}) \right],
\end{aligned}$$

Mut

Mig(4.35)

Coa

État de l'échantillon au moment t

Transition entre 2 états  
ancestraux :  
Coalescence  
Mutation  
migration

État de l'échantillon au moment t' (=t+1 événement)

$$\Pr(n_t | P) = f(n_t) \cdot \sum_{n_{t'}} [IS(n_t \rightarrow n_{t'} | P) \cdot \Pr(n_{t'} | P)]$$

$$p(\mathbf{n}) = w(\mathbf{n}) \left( \sum_{a,i,j:n_{aj}>0,j \neq i} \lambda_{aij}(\mathbf{n}) p(\mathbf{n} - \mathbf{e}_{aj} + \mathbf{e}_{ai}) \right. \\ \left. + \sum_{a,b,b \neq a,i:n_{bi}>0} I_{abi}(\mathbf{n}) p(\mathbf{n} - \mathbf{e}_{bi} + \mathbf{e}_{ai}) \right. \\ \left. + \sum_{a,j:n_{aj}>1} \mu_{aj}(\mathbf{n}) p(\mathbf{n} - \mathbf{e}_{aj}) \right)$$

Mut

Mig

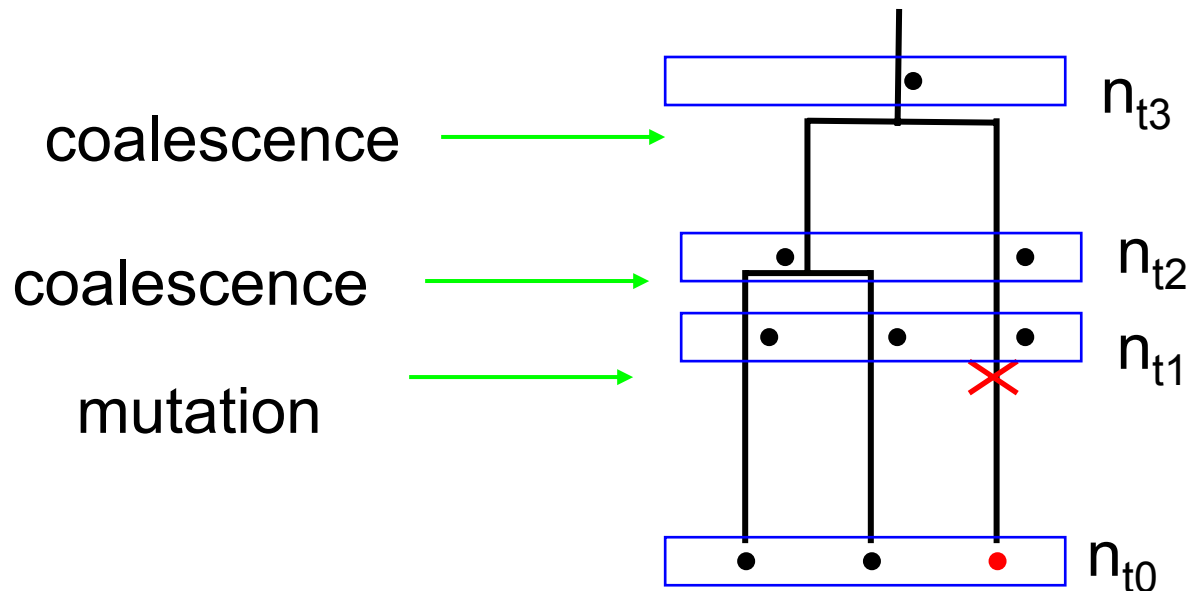
Coa

avec  $\{\lambda; I; \mu; \} = IS$      $w(n) = f(n)$

# Griffiths et coll. : Chaînes de Markov absorbantes et de l'Importance Sampling (IS)

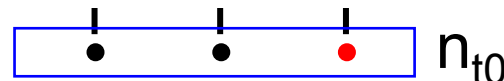
- La récurrence de base

$$\Pr(n_t|P) = f(n_t) \cdot \sum_{n_{t'}} [IS(n_t \rightarrow n_{t'}|P) \cdot \Pr(n_{t'}|P)]$$



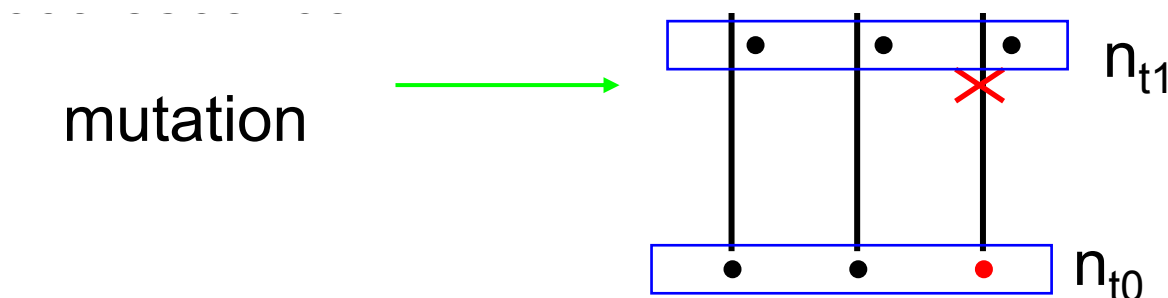
# Construction de l'arbre de coalescence en IS

1. État de départ ( $t=t_0$ )=configuration de l'échantillon  $n_{t_0}$
2. On tire au hasard l'événement suivant (=coa ou mig ou mut) parmi tout les évènements possibles avec Proba de transition =  $IS (n_t \rightarrow n_{t'})$   
→ Nouvel état  $n_{t'}$ ,  
on calcul le poids  $f(n_{t'})$
3. On recommence 2. jusqu'à ce qu'on ai **un unique gène** ancetre =**MRCA**.



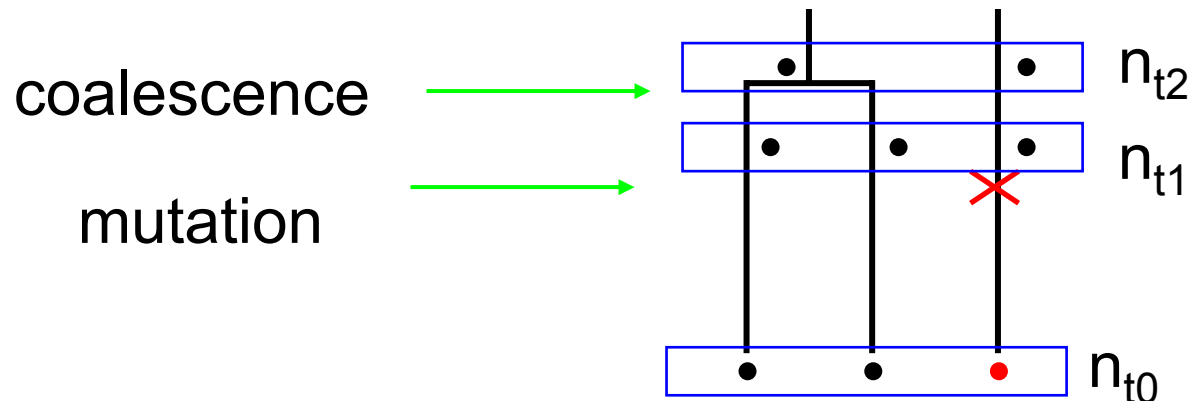
# Construction de l'arbre de coalescence en IS

1. État de départ ( $t=T_0$ )=configuration de l'échantillon  $n_{t_0}$
2. On tire au hasard l'événement suivant (=coa ou mig ou mut) parmi tous les événements possibles avec Proba de transition =  $IS (n_t \rightarrow n_{t'})$   
→ Nouvel état  $n_{t'}$ ,  
on calcul le poids  $f(n_{t'})$
3. On recommence 2. jusqu'à ce qu'on ait un unique gène ancêtre = **MRCA**.



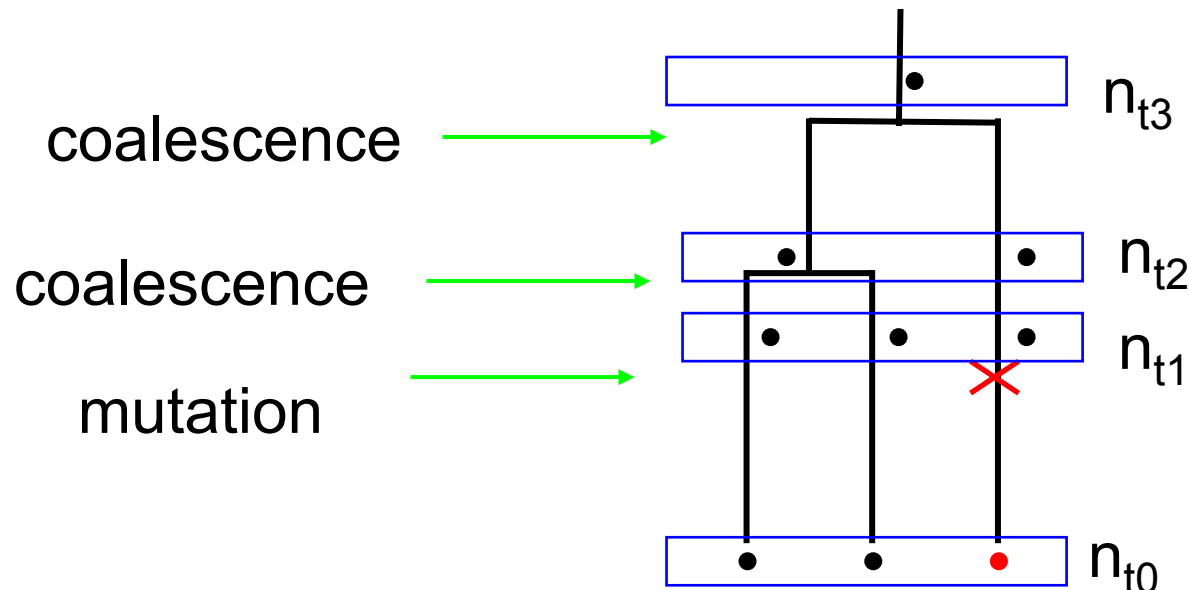
# Construction de l'arbre de coalescence en IS

1. État de départ ( $t=T_0$ )=configuration de l'échantillon  $n_{t_0}$
2. On tire au hasard l'événement suivant (=coa ou mig ou mut) parmi tous les événements possibles avec Proba de transition =  $IS (n_t \rightarrow n_{t'})$   
→ Nouvel état  $n_{t'}$ ,  
on calcul le poids  $f(n_{t'})$
3. On recommence 2. jusqu'à ce qu'on ait un unique gène ancêtre = **MRCA**.



# Construction de l'arbre de coalescence en IS

1. État de départ ( $t=T_0$ )=configuration de l'échantillon  $n_{t_0}$
2. On tire au hasard l'événement suivant (=coa ou mig ou mut) parmi tout les évènements possibles avec Proba de transition =  $IS (n_t \rightarrow n_{t'})$   
→ Nouvel état  $n_{t'}$ ,  
on calcul le poids  $f(n_{t'})$
3. On recommence 2. jusqu'à ce qu'on ai **un unique gène** ancetre =**MRCA**.



# **Griffiths et coll. : Chaînes de Markov absorbantes et de l'Importance Sampling (IS)**

- La récurrence de base

$$\Pr(n_t | P) = f(n_t) \cdot \sum_{n_{t'}} [IS(n_t \rightarrow n_{t'} | P) \cdot \Pr(n_{t'} | P)]$$

- Construction d'un arbre de coalescence (i.e. une généalogie)

$$\rightarrow \hat{\Pr}(n_{t_0} | P) = \prod_{j=t_0}^{TMRC A} f(n_j)$$



# Griffiths et coll. : Chaînes de Markov absorbantes et de l'Importance Sampling (IS)

- La récurrence de base

$$\Pr(n_t|P) = f(n_t) \cdot \sum_{n_{t'}} [IS(n_t \rightarrow n_{t'}|P) \cdot \Pr(n_{t'}|P)]$$

- Construction d'un arbre de coalescence (i.e. une généalogie)

→

$$\hat{\Pr}(n_{t0}|P) = \prod_{j=t0}^{TMRCA} f(n_j)$$

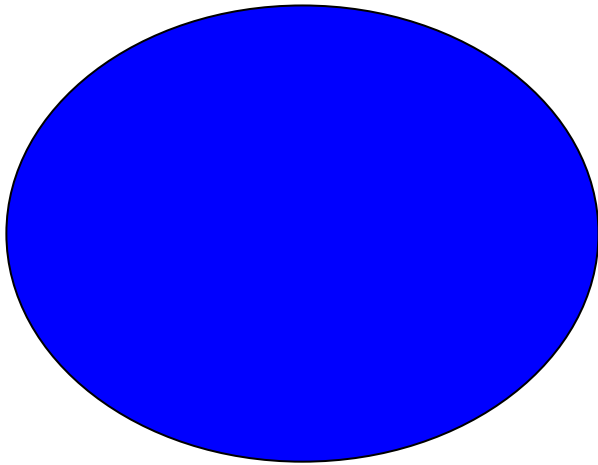
- On construit **K** arbres et on prend la moyenne sur tout les arbres

$$\hat{\Pr}(n_{t0}|P) \approx \frac{1}{K} \sum_{k=1}^K \left( \prod_{j=t0}^{TMRCA} f(n_{kj}) \right) \quad \text{cf : } L(P|D) = \frac{1}{K} \sum_{k=1}^K \Pr(D|G_k; P)$$

# Aparté : Modèles démographiques de populations :

## 1 la population panmictique

➤ Le plus simple et le plus utilisé



Une seule population dans laquelle tous les individus se reproduisent au hasard et ont la même valeur reproductive (même nombre de descendants à chaque génération)

Simple car un seul paramètre :

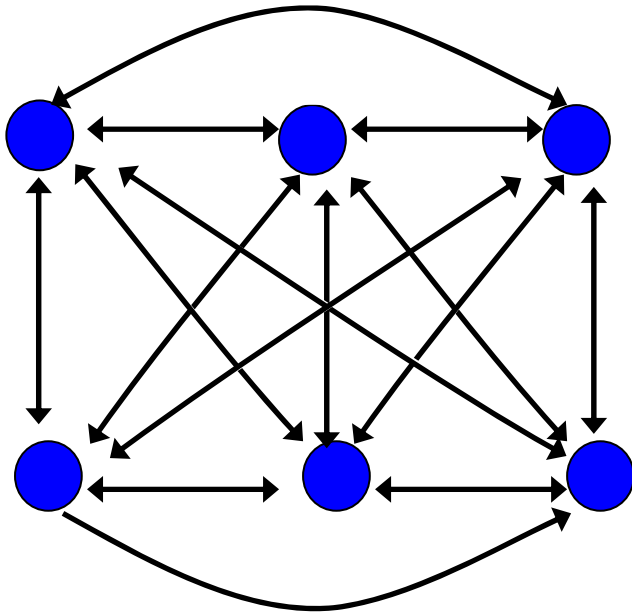
$N (\theta = 4 * N * \mu)$  = taille de la population

Problème principal : ne prend pas compte de ce qui se passe autour (i.e. autres populations échangeant des migrants)

Bon modèle pour études théoriques mais généralement pas pour l'estimation précise de paramètres démographiques

# Modèles démographiques de populations : Le modèle en îles

## ➤ Wright (1931,1937)



Simple car homogénéité réduit à 3 le nombre de paramètres :

$d$  = nombre de sous-populations (ou  $\infty$ )

$N$  ( $\theta = 4 * N * \mu$ ) = taille sous-populations

$m$  ( $\gamma = 4 * N * m$ ) = taux de migration

On a alors la fameuse relation:

$F_{st} = 1 / (1 + 4Nm)$  en nombre d'iles infini

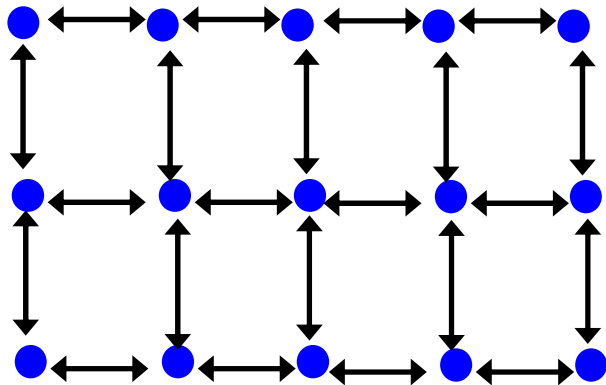
Problème principal : Migration indépendante de la distance entre sous-populations -> pas très réaliste

Bon modèle pour études théoriques mais pas pour estimation précise de paramètres démographiques

## 2. Modèles démographiques de populations structurées :

### 2.2 Le modèle de migration par pas (stepping stone)

➤ Kimura (1953), Malécot (1959), Kimura and Weiss (1964)



Aussi très simple, Peu de paramètres

$d$  = nombre de sous-populations

$N$  ( $\theta = 4 * N * \mu$ ) = taille sous-populations

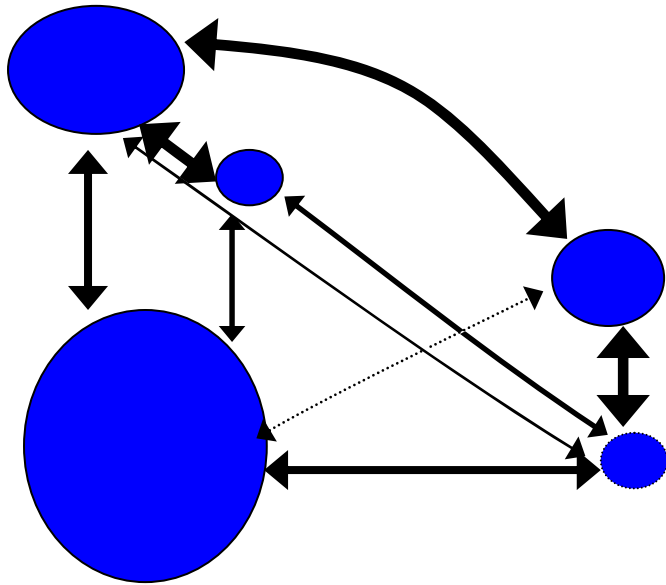
$M$  ( $\gamma = 4 * N * m$ ) = taux de migration

Migration seulement entre sous-populations adjacentes  
-> pas très réaliste mais 1<sup>er</sup> modèle analysable avec dispersion localisé dans l'espace

## 2. Modèles démographiques de populations structurées :

### 2.3 Le modèle avec matrice de migration libre

Modèle le plus général



**Non homogène -> Beaucoup de paramètres**

**$d$  = nombre de sous-populations**

**$\{N_1 (\theta_1 = 4 * N_1 * \mu), \dots, N_d\}$  = taille des sous-populations**

**$\{m_{ij} (\gamma_{ij} = 4 * N_i * m_{ij})\}$  = taux de migration entre paires de populations**

Très réaliste mais trop de paramètres -> **problème pour l'estimation**

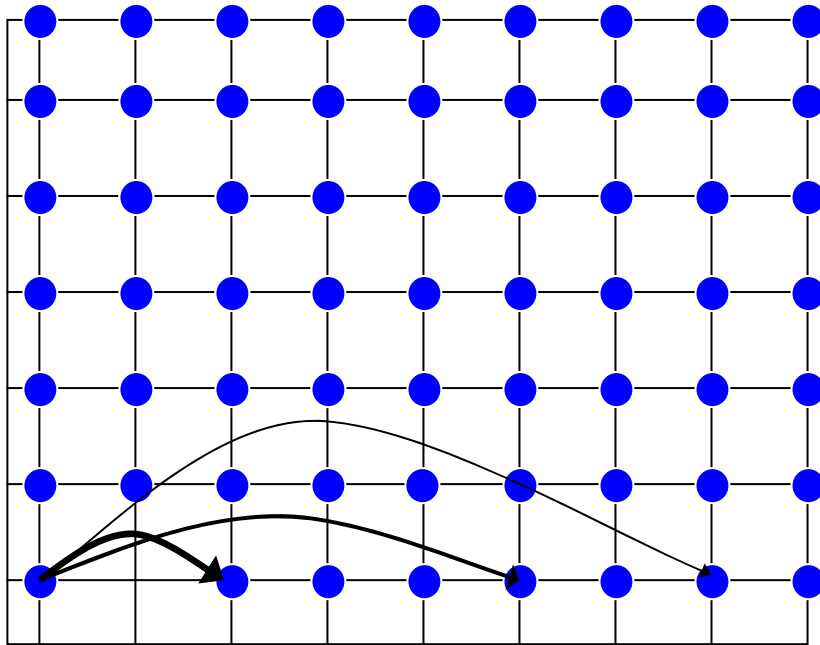
Homogénéisation -> modèle en îles, stepping stone, IBD, ...

## 2. Modèles démographiques de populations structurées :

### 2.4 Le modèle d'isolement par la distance

Dispersion limitée dans l'espace  $\leftrightarrow$  2 individus ont plus de chance de se reproduire ensemble si ils sont proches géographiquement

Endler 1977 (revue biblio): la majorité des espèces ont une dispersion localisé

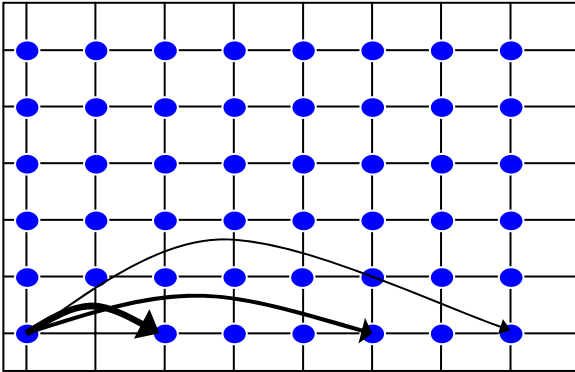


Migration fonction de la distribution de dispersion :

## 2. Modèles démographiques de populations structurées :

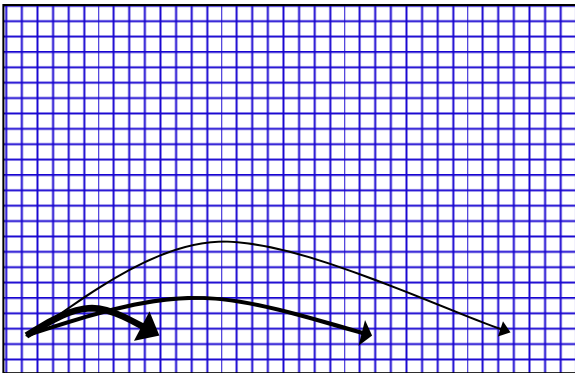
### 2.4 Les modèle d'isolement par la distance

2 modèles en fonction du type de distribution des organismes dans le paysage :



Population en dèmes

Chaque nœud du réseau correspond à une sous population panmictique



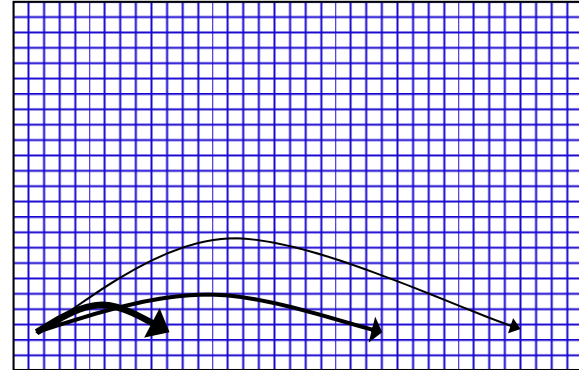
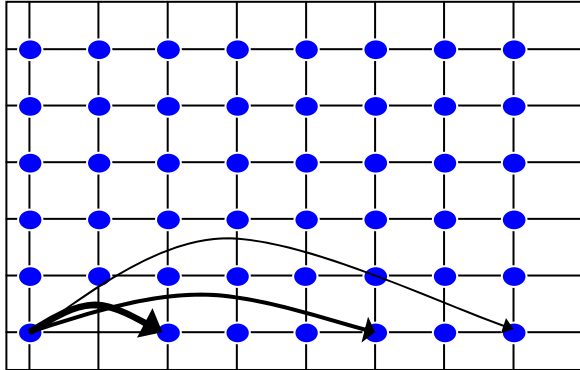
Population "continue" en réseaux

Chaque nœud du réseau correspond à 1 individu

## 2. Modèles démographiques de populations structurées :

### 2.4 Les modèle d'isolement par la distance

2 modèles en fonction du type de distribution des organismes dans le paysage :



Dans les 2 cas, homogénéité spatiale :

Dèmes de taille identique ou densité d'individus identique sur tout le réseau  
Distribution de dispersion identique en tout point du réseau

-> peu de paramètres (2-3) :

$\sigma^2$  = Carré moyen de la distance de dispersion parent-descendant  
= inverse de "force de l'isolement par la distance"

$N$  ( $\theta = 4 * N * \mu$ ) = taille des sous-populations quand structure en dèmes  
ou  $D$  = densité d'individu quand population continue

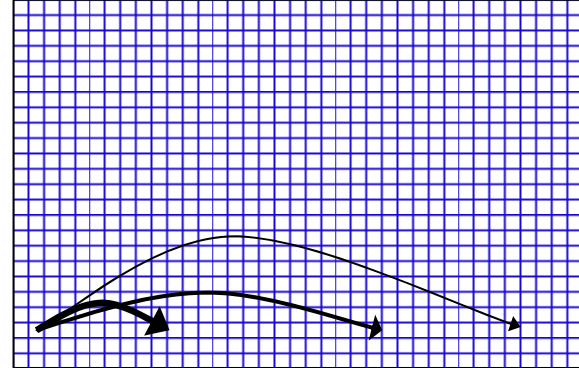
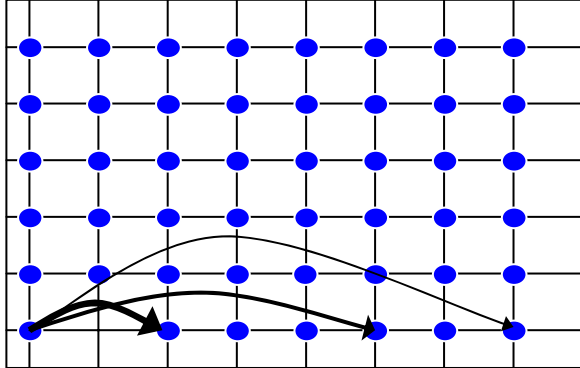
+ -  $m$  ( $\gamma = 4 * N * m$ ) = taux de migration (emmigration total d'un dème)



## 2. Modèles démographiques de populations structurées :

### 2.4 Les modèle d'isolement par la distance

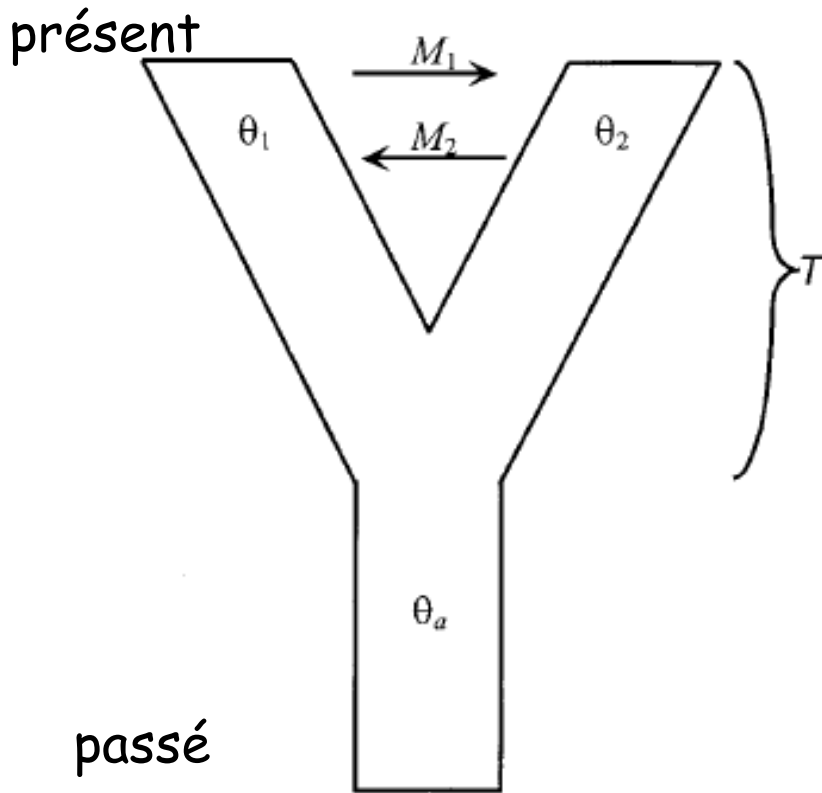
2 modèles en fonction du type de distribution des organismes dans le paysage :



Modèle assez réaliste et très bon pour l'estimation de paramètres démographiques pour de multiples raisons non explicitées ici...

## 2. Modèles démographiques de populations structurées :

### 2.4 Le modèle de divergence avec migration (IM : isolation with Migration)



Réaliste (histoire commune des pops), peu de paramètres dans la version simple a 2 pops :

$N_1$  et  $N_2$  ( $\theta_1$  et  $\theta_2$ ) = taille sous-populations actuelles

$N_a$  ( $\theta_a$ ) = taille pop ancestrale

$m_1$  et  $m_2$  ( $\gamma_1$  et  $\gamma_2$ ) = taux de migration

$T$  ( $t=T/2N$ ) = temps de divergence

Bon modèle pour la divergence de population/espèces

Pb : avec plus de 2 sous population on augmente très rapidement le nbre de paramètres (non utilisable en pratique)<sup>74</sup>

# ***Griffiths et coll. : Chaînes de Markov absorbantes et de l'Importance Sampling (IS)***

## **Historique (quasi-exhaustif)**

- Griffiths et Tavaré 1994 : 1<sup>er</sup> algo IS pour 1 population panmictique (données génotypiques tout modèles de mutations)
- Nath et Griffiths 1996 : adaptation de GT94 pour un modèle en îles (données génotypiques tout modèles de mutation)
- Bahlo et Griffiths 2000 : adaptation aux données de type séquences (mais que Infinite Site Model) -> logiciel GENETREE
- Stephens & Donnelly 2000 : bien meilleure fonction d'IS pour une population panmictique (tout modèle de mutation mais plus efficace avec PIM)
- De Iorio & Griffiths 2004 : généralisation théorique de la fonction d'IS de S&D2000 pour différents modèles démo dont des populations structurées (+- facile à implémenter selon les modèles démo et mutationnel)

# Temps de calcul et complexité des modèles avec l'algorithme de Nath et Griffiths (1996)

Nbre d'itérations et temps nécessaire pour estimer correctement la vraisemblance d'un échantillon en 1 points de l'espace des paramètres

Complexité = nbre de pops x nbre d'états alléliques possibles

10 pop

→ ?

4 pop

30 allèles

?

4 allèles

3 H

2 pop

4 allèles

2 pop

2 allèles

1H30 (1 GHz)

10 min

600000  
500000  
400000  
300000  
200000  
100000  
0

0

5

10

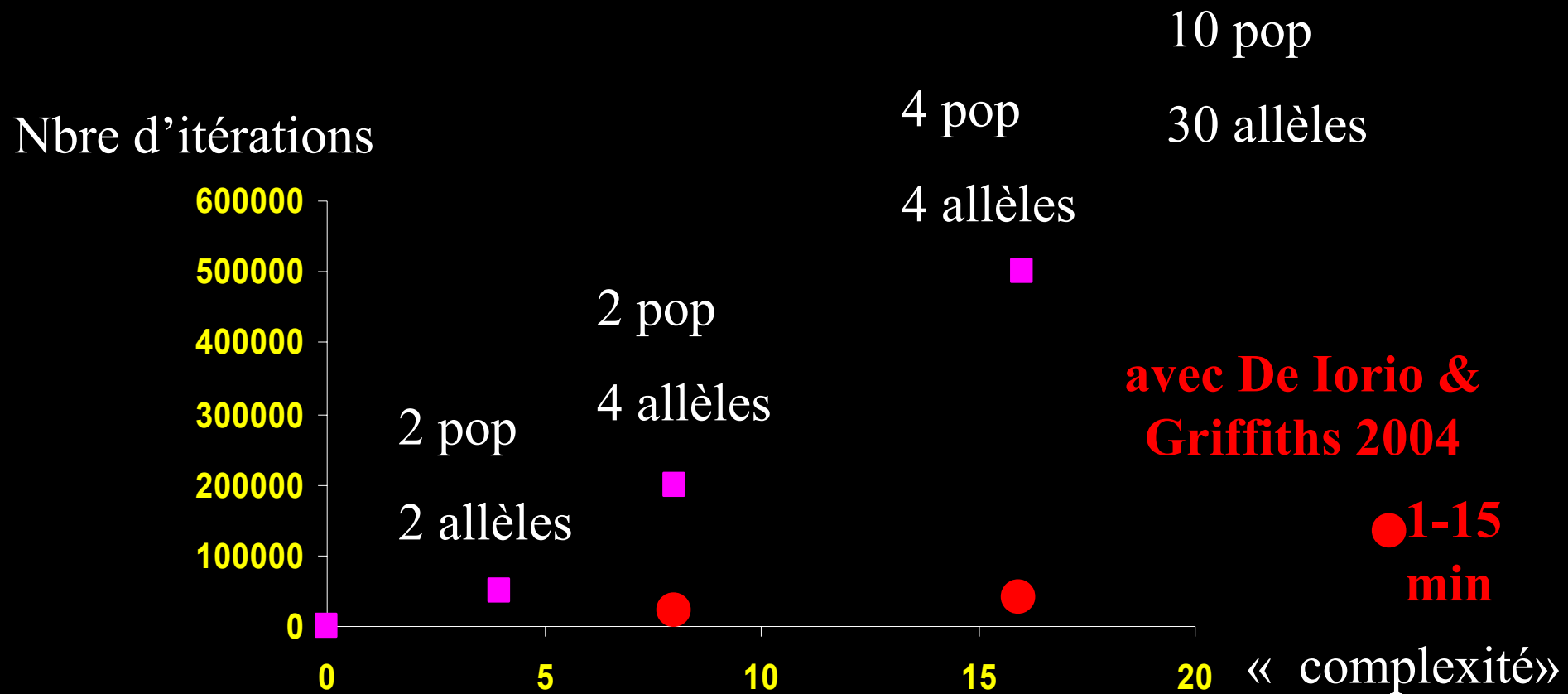
15

20

« complexité »  
76

Trop lent pour être utilisable en pratique

# Temps de calcul et complexité des modèles avec les meilleures versions de De Iorio & Griffiths 2004



→ Beaucoup plus performant, utilisable en pratique

**...mais ces très bonnes performances sont valable  
uniquement pour mutations indépendantes du type  
parental (PIM/KAM) et modèle de migration  
simples (en îles)... plus complexe pour d'autres  
modèles  
mutationnels et démographiques**

En PIM et en population panmictique, la nouvelle fonction d'IS permet d'avoir la vraisemblance exacte avec un seul arbre = fonction IS optimale

Quand on complexifie le modèle démographiques mais toujours en PIM ce n'est plus la fonction optimale:

limite1 = résolution d'un système d'équation linéaire de dimension Nbre pop x Nbre d'états alléliques

Limite2 = plus le modèle est complexe plus il faut d'arbres (entre 5 et 500)

Pour d'autre modèles mutationnels que PIM beaucoup plus difficile

# ***Griffiths et coll. : Chaînes de Markov absorbantes et de l'Importance Sampling (IS)***

## **Historique (quasi-exhaustif) suite**

- Delorio, Leblois, Griffiths & Rousset 2005 : adaptation pour 2 population avec migration et mutation par pas SMM (microsatellites)
- Rousset & Leblois 2007 : adaptation pour isolement par la distance en une dimension et mutation PIM/KAM -> logiciel MIGRAINE

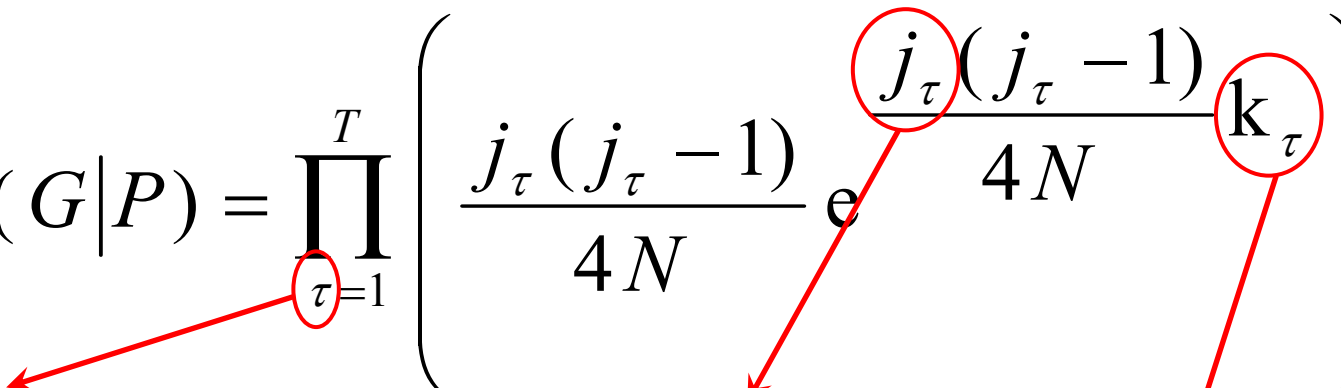
## **Bientôt disponible (=adapté et codé dans MIGRAINE mais non testé et non publié)**

- Modèle à  $N < 3-4$  pops avec migration "libre" (modèle "Migration matrix") et mutation PIM/KAM
- Modèle de divergence de 2-3 pops avec flux de gènes (modèle "Isolation with Migration") mutation PIM/KAM ou SMM

**A faire (PostDoc???) adapter les algo IS pour analyser des séquences dans tous ces modèles démo**

# Felsenstein et coll.: Monte Carlo par Chaînes de Markov (MCMC)

- Probabilité d'une généalogie sachant les paramètres démographiques du modèle:  $N, [N_i m_{ij} \text{ si population structurée}]$   
exemple pour une pop panmictique

$$\Pr(G|P) = \prod_{\tau=1}^T \left( \frac{j_{\tau}(j_{\tau}-1)}{4N} e^{-\frac{j_{\tau}(j_{\tau}-1)}{4N} k_{\tau}} \right)$$


Produit sur tous les évènements « démographiques » (coalescence ou migration si pop structurée) de la généalogie

Nombre de lignées avant l'évènement

Intervalle de temps entre cet évènement et le précédent



# Felsenstein et coll.: Monte Carlo par Chaînes de Markov (MCMC)

- Probabilité d'une généalogie sachant les paramètres démographiques du modèle ( $N, m_{ij}$ )

$$\Pr(G|P) = \prod_{\tau=1}^T \left( \frac{j_{\tau}(j_{\tau}-1)}{4N} e^{\frac{j_{\tau}(j_{\tau}-1)}{4N} k_{\tau}} \right)$$

- Probabilité de l'échantillon sachant la généalogie et les paramètres mutationnels ( $\mu, P_{\text{mut}}$  matrice de mutation)

$$\Pr(D|G) = \prod_{b=1}^B \left( (P_{\text{mut}})^{i_b} \frac{(\mu L_b)^{i_b}}{i_b!} e^{-\mu L_b} \right)$$

Produit sur toutes les branches de l'arbre

Nombre de mutation sur la branche b

Loi de poisson pour la probabilité d'avoir i mutation sur un intervalle de temps Lb

Longueur de la branche b

# Felsenstein et coll.: Monte Carlo par Chaînes de Markov (MCMC)

- Probabilité d'une généalogie sachant les paramètres démographiques du modèle  $(N, m_{ij})$

$$\Pr(G|P) = \prod_{\tau=1}^T \left( \frac{j_{\tau}(j_{\tau}-1)}{4N} e^{\frac{j_{\tau}(j_{\tau}-1)}{4N}} k_{\tau} \right)$$

- Probabilité de l'échantillon sachant la généalogie et les paramètres mutationnels

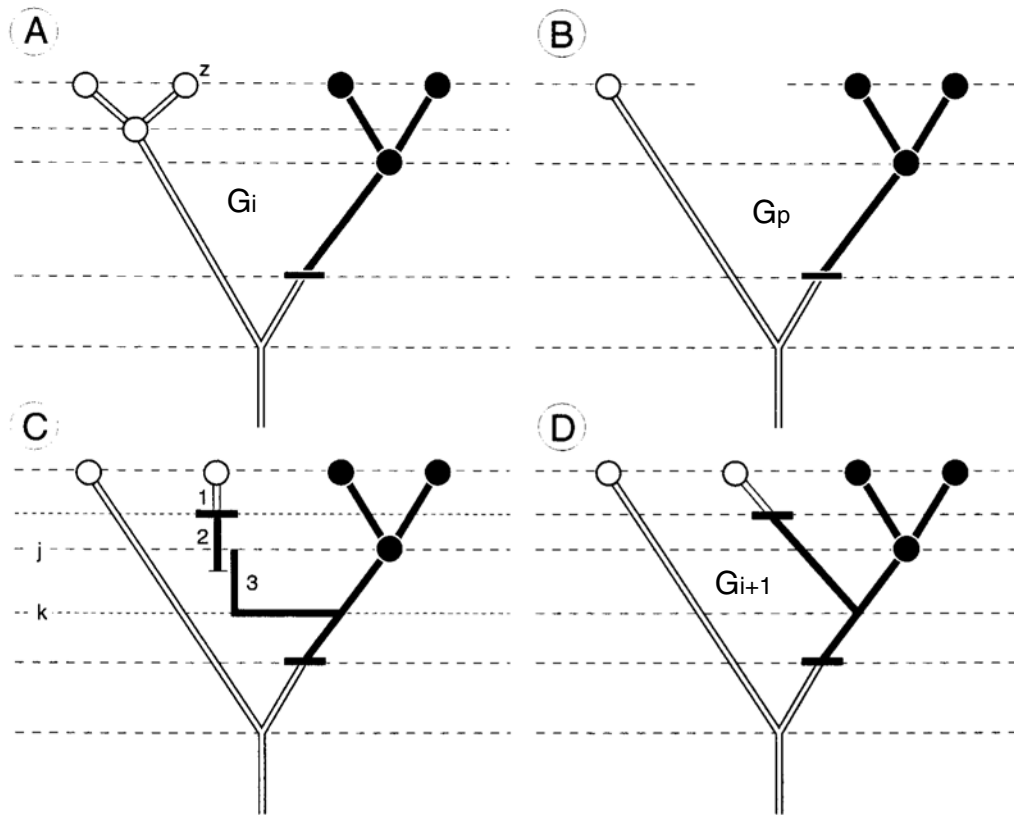
$$\Pr(D|G) = \prod_{b=1}^B \left( (P_{mut})^{i_b} \frac{(\mu L_b)^{i_b}}{i_b!} e^{-\mu L_b} \right)$$

- Par définition

$$L(P|D) \approx \frac{1}{K} \sum_{k=1}^K \Pr(D|G_k; P) \approx \frac{1}{K} \sum_{k=1}^K \Pr(D|G_k) \Pr(G_k|P)$$

# L'échantillonnage des généalogies en utilisant des MCMC

1. Construction d'un arbre « probable » de départ à partir de l'échantillon (UPGMA, neighbour joining)
2. Construction d'une nouvelle généalogie par délétion-reconstruction d'un bout

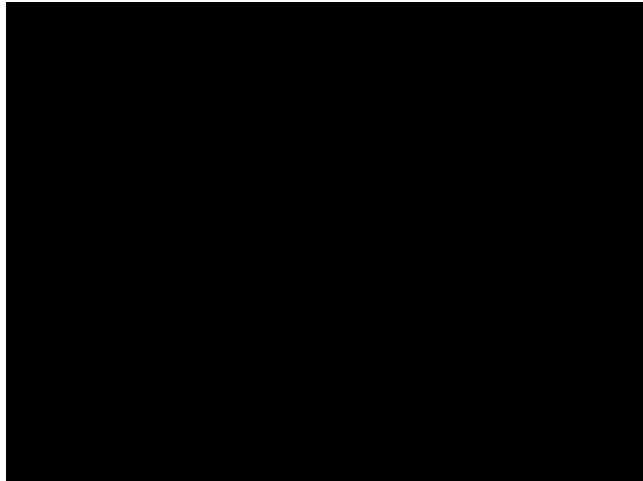


# L'échantillonnage des généalogies utilisant des MCMC

1. Construction d'un arbre « probable » de départ à partir de l'échantillon (UPGMA, neighbour joining)
  2. Construction d'une nouvelle généalogie par délétion-reconstruction d'un bout
  3. Acceptation ou non de la nouvelle généalogie  
(critère de Metropolis-Hasting proportionnel à  $\frac{\Pr(D|G_{i+1})}{\Pr(D|G_i)}$  )
1. On recommence N fois 2. et 3.

# L'échantillonnage des généalogies utilisant des MCMC

1. Construction d'un arbre « probable » de départ à partir de l'échantillon (UPGMA, neighbour joining)
  2. Construction d'une nouvelle généalogie par délétion-reconstruction d'un bout
  3. Acceptation ou non de la nouvelle généalogie (critère de Metropolis-Hasting proportionnel à  $\frac{\Pr(D|G_{i+1})}{\Pr(D|G_i)}$  )
1. On recommence N fois 2. et 3.



→ **!! Donne des arbres corrélés !!**

# Felsenstein et coll.: Monte Carlo par Chaînes de Markov (MCMC)

L'exploration de l'espace des paramètres se fait en même temps que l'échantillonnage des généalogies

i.e. une étape (update) de la MCMC = soit changement de généalogie soit changement de valeur d'un des paramètres

Beaucoup de variantes de cet algorithme, les différences étant essentiellement dans l'exploration de l'espace des paramètres et des généalogies (quels updates avec quelles probabilités)...

L'analyse se fait ensuite soit dans un cadre bayésien soit en maximum de vraisemblance.

# Felsenstein et coll.: Monte Carlo par Chaînes de Markov (MCMC)

Historique (non-exhaustif)

Premières publications en population panmictique

FELSENSTEIN, J., 1981 Evolutionary trees from DNA sequences: a maximum likelihood approach. *J. Mol. Evol.* **17**: 368–376.

Kuhner, M., Yamato, J. and Felsenstein, J. (1995) Estimating effective population size and mutation rate from sequence data using Metropolis-Hastings sampling. *Genetics* **140**: 1421–1430.

Kuhner, M., Yamato, J. and Felsenstein, J. (1998) Maximum likelihood estimation of population growth rates based on the coalescent. *Genetics* **149**: 429–434.

→ logiciel FLUCTUATE (maintenant LAMARC : pop size, growth rate, migration rates and recombination rates)

# Felsenstein et coll.: Monte Carlo par Chaînes de Markov (MCMC)

Historique (non-exhaustif)

En population panmictique mais avec recombinaison :

Kuhner, M., Yamato, J. and Felsenstein, J. (2000) Maximum likelihood estimation of recombination rates from population data. *Genetics* **156**: 1393–1401.

Fearnhead, P. and Donnelly, P. (2001) Estimating recombination rates from population genetic data. *Genetics* **159**: 1299–1318.

Fearnhead, P. and Donnelly, P. (2002) Approximate likelihood methods for estimating local recombination rates. *J. Royal Statist. Soc. B* **64**: 657–680.

→ logiciel RECOMBINE (maintenant LAMARC : pop size, growth rate, migration rates and recombination rates)

Autre algorithme MCMC (proche) :

Nielsen, R. (2000) Estimation of population parameters and recombination rates from single nucleotide polymorphisms. *Genetics* **154**: 931–942.



# Felsenstein et coll.: Monte Carlo par Chaînes de Markov (MCMC)

En populations structurées :

BEERLI, P. & FELSENSTEIN, J. (1999) Maximum likelihood estimation of migration rates and effective population numbers in two populations using a coalescent approach. *Genetics* **152** : 763–773.

BEERLI, P. & FELSENSTEIN, J. (2001) Maximum likelihood estimation of a migration matrix and effective population sizes in  $n$  subpopulations by using a coalescent approach. *Proceedings of the National Academy of Sciences of the U.S.A.* **98** : 4563–4568.

→ logiciel MIGRATE (modèle Matrice de migration), marche pas bien sauf quand 2-3 pops

# Felsenstein et coll.: Monte Carlo par Chaînes de Markov (MCMC)

Premiers développement en divergence simple puis avec migration :

NIELSEN, R., 1998 Maximum likelihood estimation of population divergence times and population phylogenies under the infinite sites model. *Theor. Popul. Biol.* **53**: 143–151.

NIELSEN, R., and J. WAKELEY, 2001 Distinguishing migration from isolation: a Markov chain Monte Carlo approach. *Genetics* **158**: 885–896.

Copyright © 2004 by the Genetics Society of America  
DOI: 10.1534/genetics.103.024182

**Multilocus Methods for Estimating Population Sizes, Migration Rates and Divergence Time, With Applications to the Divergence of *Drosophila pseudoobscura* and *D. persimilis***

Jody Hey<sup>\*,1</sup> and Rasmus Nielsen<sup>†</sup>

→ logiciel IM (modèle IM 2 populations), semble bien marcher mais temps de calculs très long quand plusieurs locus

# Felsenstein et coll.: Monte Carlo par Chaînes de Markov (MCMC)

Dernière amélioration :

**Integration within the Felsenstein equation for improved Markov chain Monte Carlo methods in population genetics**

Jody Hey<sup>†‡</sup> and Rasmus Nielsen<sup>§</sup>

→ logiciel IMa (modèle IM 2 populations), beaucoup plus rapide que IM mais pas encore bien testé

Pas encore très bien compris mais le MCMC ne fait plus que l'exploration des généalogies, l'exploration de l'espace des paramètres se fait analytiquement (je pense par approximation...)

# Avantages et inconvénients des deux approches

## ➤ MCMC

beaucoup plus avancé dans les différents modèles démographiques et mutationnels possibles (i.e. plus flexible)

mais il y a souvent des problèmes de convergence et de mixage, de corrélation des généalogies et de paramétrage des MCMC assez complexes, surtout quand nombreux paramètres

Difficilement testable à cause des longs temps de calculs

## ➤ IS

Bien meilleure efficacité de l'échantillonnage avec les améliorations de SD2001 (1 arbre = vraisemblance exacte), DG2004 et suivantes

Temps de calculs beaucoup plus court et on a la "vrai" surface de vraisemblance → Méthodes plus facilement testables

Mais pas très flexible, l'adaptation des algorithmes à des modèles spécifique est plus complexe

Pb quand plus de 4-5 paramètres pour extrapoler la surface de vraisemblance à partir de peu de points (= krigeage)

# Conclusions

- Fortes potentialités : plus d'info, plus souple
- Mais au stade actuel, peu d'algorithmes performants pour l'estimation de taux de migration
- Quelques améliorations possibles :
  - Limiter le nombre de paramètres à estimer (MIGRATE)
  - Limiter les temps de calculs (IS)
- Bon outil pour l'étude d'une population panmictique.
- Améliorations en cours pour les populations structurées et modèles de divergence (encore incertain pour les modèles à plus de 4-5 paramètres...)
- Adaptation au cas par cas possible pour certaines études pas trop complexes (notamment avec ABC)

**Les résultats d'application sur des données réelles complexes doivent être interprétés avec prudence**

## 2 populations, mutation par pas (SMM)

De Iorio, Griffiths, Leblois, Rousset, 2005 TPB

➤ Cas spécial de De Iorio & Griffiths (2004a):  
résolu par transformée de Fourier

- Résultats préliminaires :
- Un jeu de données réel (renard)
  - Quelques simulations



# Australian Red Fox

(Lade *et al.* 1996)



- DATA :
  - 2 populations (Island, Mainland)
  - 7 microsatellites
- MODEL :
  - Single step mutation (SMM)
  - 3 parameter estimation ( $\theta=4N\mu$ ,  $4N_M m_{MI}$ ,  $4N_I m_{IM}$ )
  - 1 million runs for 30 parameter sets ( $\theta_i$ ,  $4N_M m_{MIi}$ ,  $4N_I m_{Ii}$ )  
(~few days on 1Ghz)

# Australian Red Fox (Lade *et al.* 1996)

## Results

- Good convergence between independent runs

- MLE :  $4N_M m_{MI} = 4.0$

$$4N_I m_{IM} = 3.0$$

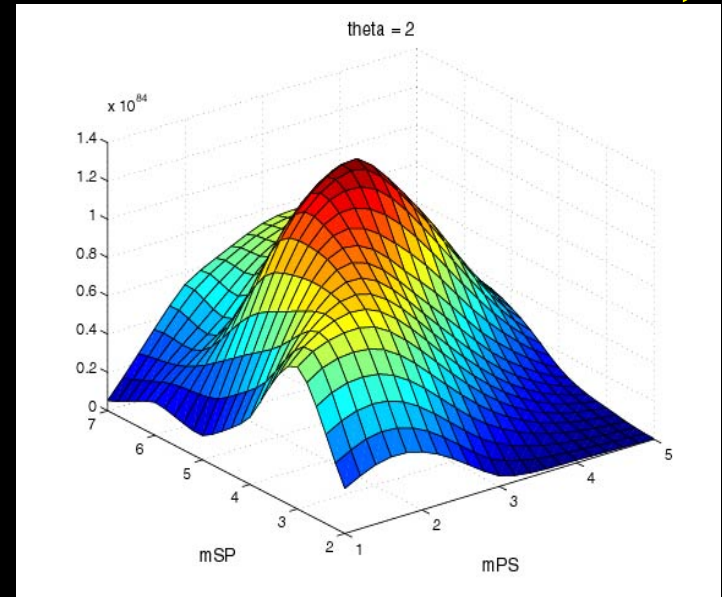
- For comparison :

$$-F_{ST} \rightarrow 4N m \sim 3.0 \quad (R_{ST} \rightarrow 4N m \sim 7.4)$$

–MIGRATE :

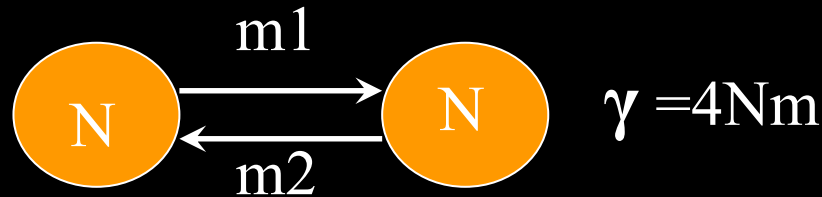
$$4N_M m_{MI} = [2.3-3.0-3.8-1.5] \quad 4N_I m_{IM} = [1.4-3.6-2.8-1.0]$$

(large variance between runs with different starting values)





# Tests par Simulation



- ✓ 2 populations (N=1000, même  $\theta=4N\mu=2.0$ )
- ✓ Migration symétrique ( $4Nm1=4Nm2=2.0$ )
- ✓ Mutation par pas (SMM)
- ✓ 30 individus pour 5 et 20 locus
- ✓ 10 jeux de données (1 mois sur 50 processeurs 1 GHz!!)

# Résultats des simulations

estimation du paramètre de migration  $\gamma = 4Nm$

➤ IS : Griffiths et al.

➤ MCMC : MIGRATE

à temps de calcul comparables

✓ 5 locus

- Biais relatif=0.6
- MSE=2.2

✓ 5 locus

- Biais relatif=2.38
- MSE=12.5

✓ 20 locus

- Biais relatif=0.5
- MSE=1.2

✓ 20 locus

- Biais relatif=0.5
- MSE=2.6

## **...à ce stade, beaucoup de problèmes persistent pour le MV...**

- Temps de calcul (IS et MCMC) très long mais amélioration récentes réduisant les temps de calculs pour IS
- Surestimation (inhérente aux méthodes?)

# ...à ce stade, beaucoup de problèmes persistent pour le MV...

- Temps de calcul (IS et MCMC)
- Surestimation (inhérente aux algorithmes?)

Il faut encore tester l'effet de:

- Nombre de populations échantillonnées vs nombre total de sous-populations (testé pour MIGRAINE, cf 2eme partie du cours)
- Processus mutationnels complexes des locus microsatellites (déviations du modèle par pas)
- Effet de fluctuations démographiques passées<sup>100</sup>

# **Alternatives : un exemple d'approche bayésienne**

## **Approximate Bayesian Computation (ABC)**

**UTILISATION DES MICROSATELLITES POUR INFERER DES HISTOIRES  
DEMOGRAPHIQUES COMPLEXES ET RECENTES :**

**LE CAS D'UNE COLONISATION INSULAIRE PAR UN  
OISEAU (*ZOPTEROPS LATERALIS*)**

***Université de Brisbane (Australie)***

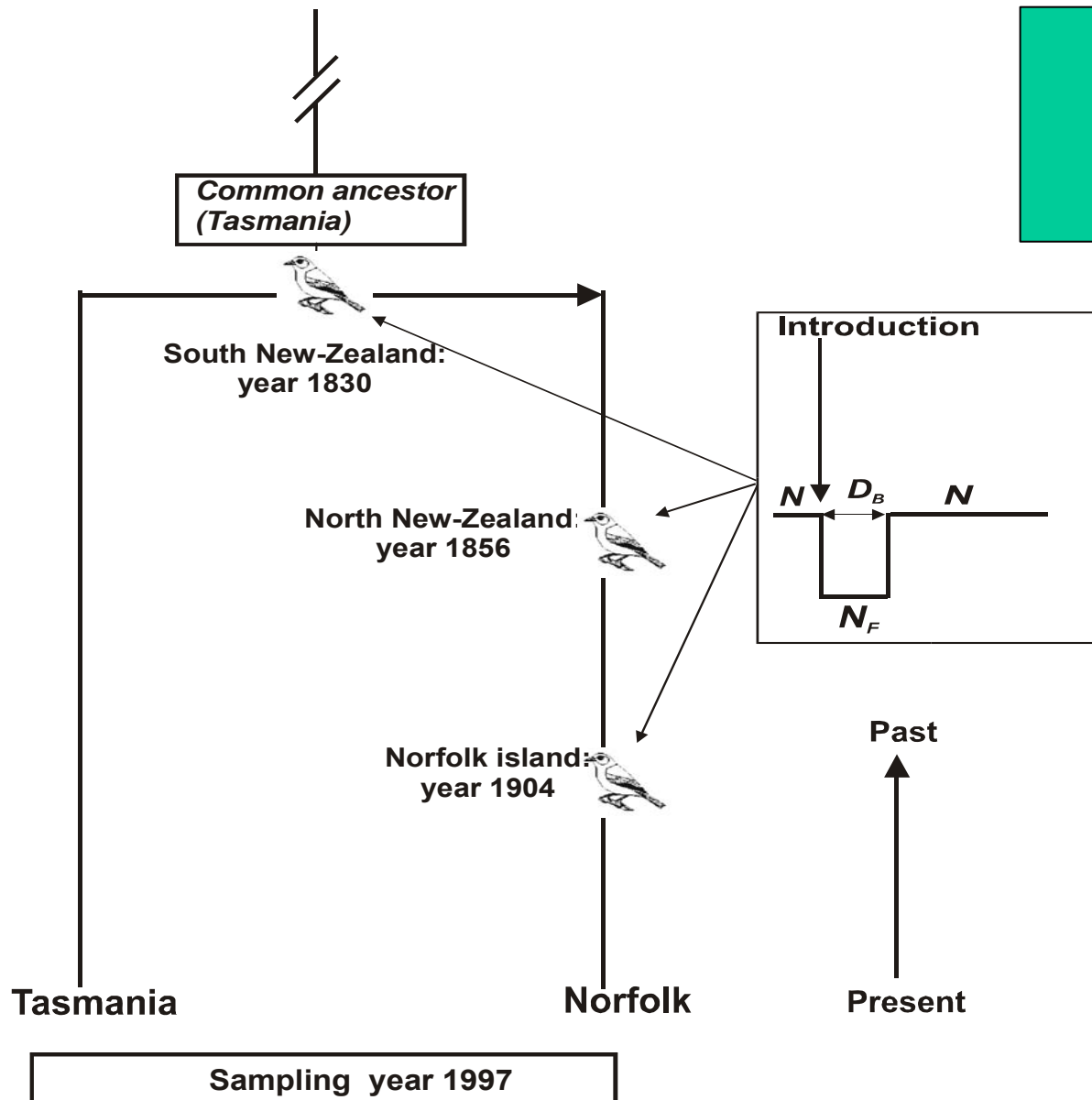
**Sonia Clegg**

**Ian Owens**

**Craig Moritz**

***CBGP, Montpellier (France)***

**Arnaud Estoup**



Modèle  
en urne

Approx.  
temps  
continu

# PROCEDURE D'ESTIMATION ABC

MODELE avec  
Paramètres  
FIXES

Paramètres avec PRIORS

-  $N$ ,  $N_f$ ,  $D_B$   
-  $\mu$ ,  $\sigma^2$  (GSM)  
→  $a^*$ ,  $H^*$ ,  $V^*$ ,  $F_{st}^*$

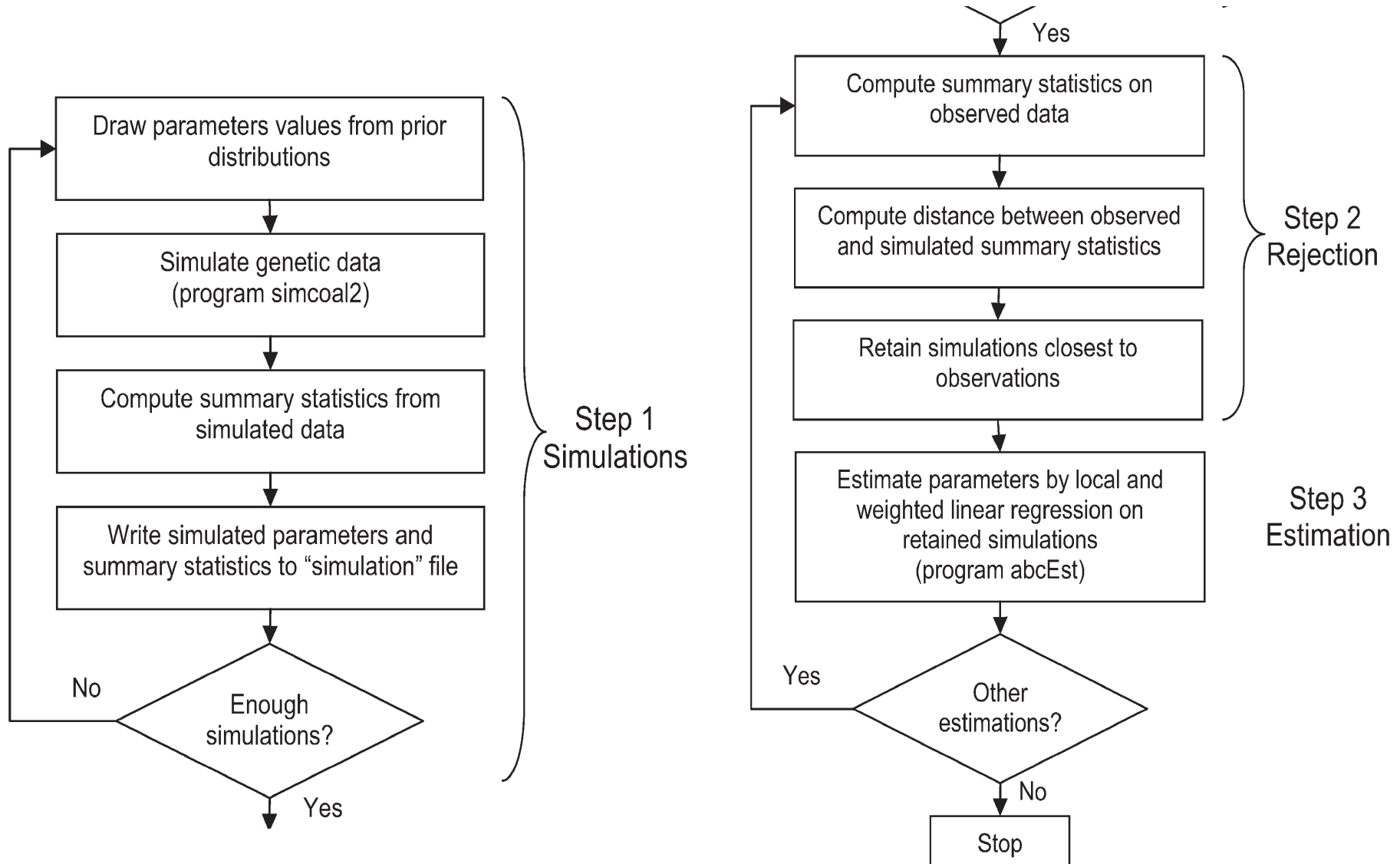
Données  
génétiques  
( $a$ ,  $H$ ,  $V$ ,  $F_{st}$ )

METHODE PAR SIMULATION = ALGORITHME DE REJET  
(Tavaré et al. 1995 ; Pritchard et al. 1999 ; Estoup et al. 2001)

Coestimation de POSTERIORIS

pour  $N$ ,  $N_f$ ,  $D_B$ ,  $\mu$ ,  $\sigma^2$

# Procedure ABC détaillée:





## Étape 1 :

### Simuler des données a l'aide de la coalescence

- Simuler des valeurs pour variables démographiques  $N$ ,  $N_f$  et  $D_B$  à partir des priors (une valeur par variable =  $*N$ ,  $*N_f$  et  $*D_B$ )
- Processus de coalescence = arbre généalogique pour une pair de pops (source + ile colonisée) avec les valeurs  $*N$ ,  $*N_f$  et  $*D_B$  pour  $m$  chromosomes (i.e. gènes,  $m$ = taille de l'échantillon) et pour chacun des 6 locus.
- Simuler les valeurs pour les variables mutationnelles  $\mu$  et  $\sigma^2$  (SOIT  $*\mu$  et  $*\sigma^2$ ) → simuler les  $m$  génotypes pour les 6 locus = mettre les mutation sur les arbres obtenus ci dessus

## Étape 2 :

### Comparer les données simulées et les vraies données à l'aide de statistiques résumées

- Calculer les statistiques résumées : nombre d'allèles, hétérozygotie, variance de taille alléliques pour la pop source et l'île + Fst entre pop source et île [ $*a_1$ ,  $*H_1$ ,  $*V_1$ ,  $*a_2$ ,  $*H_2$ ,  $*V_2$  et  $*F_{ST}$ ] à partir des génotypes simulés
- Si pour toutes les statistiques résumées on a  $|*stat-stat\_données| < \delta$  alors on enregistre les valeurs  $*N$ ,  $*N_f$ ,  $*D_B$ ,  $*\mu$  et  $*\sigma^2$
- recommencer à partir de 1 jusqu'à obtenir un échantillon de 1000 valeurs pour chaque paramètre  $N$ ,  $N_f$ ,  $D_B$ ,  $\mu$  et  $\sigma^2$  = distribution a posteriori (i.e. prenant en compte l'information des données)

**Taux d'acceptation =  $[1/25000 - 1/50000]$  POUR  $\delta=0.12$**

**→ beaucoup de généalogies a simuler ( $\approx 50\,000\,000$ )**

## Analyse des résultats

**→ COMPARER  
LES  
DISTRIBUTIONS  
PRIORS ET  
POSTERIEURS**

