Phylogenetic Tree Construction using Sequential Monte Carlo Algorithms on Posets

Liangliang Wang
Western University, Canada

Joint work with
Alexandre Bouchard-Côté
Arnaud Doucet

Recent Advances in SMC
Sep 19-21, 2012

Outline

Phylogenetics

Example: what are the evolutionary relationships among these Cichlid fishes?

(a) Chalinochromis popelini

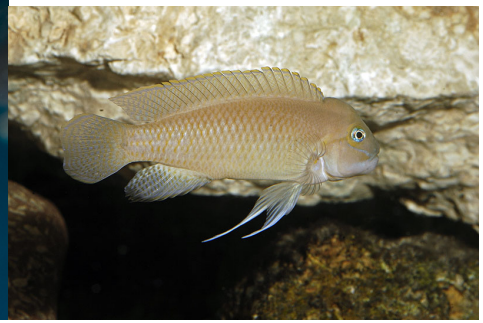
(b) Julidochromis marlieri

(c) Lamprologus callipterus

(d) Lepidiolamprologus elongatus

(e) Neolamprologus brichardi

(f) Neolamprologus tetracanthus



(g) Telmatochromis temporalis

Data: $\mathcal{Y}$

- Biological sequences of a set of species
  - e.g. a DNA sequence is a string of characters from the set of four nucleotides $\{A, C, G, T\}$.

Data: $\mathcal{Y}$

- Biological sequences of a set of species
  - e.g. a DNA sequence is a string of characters from the set of four nucleotides $\{A, C, G, T\}$.
- An example of aligned DNA sequences.
  - Nucleotides in the same column were obtained from a shared ancestral nucleotide

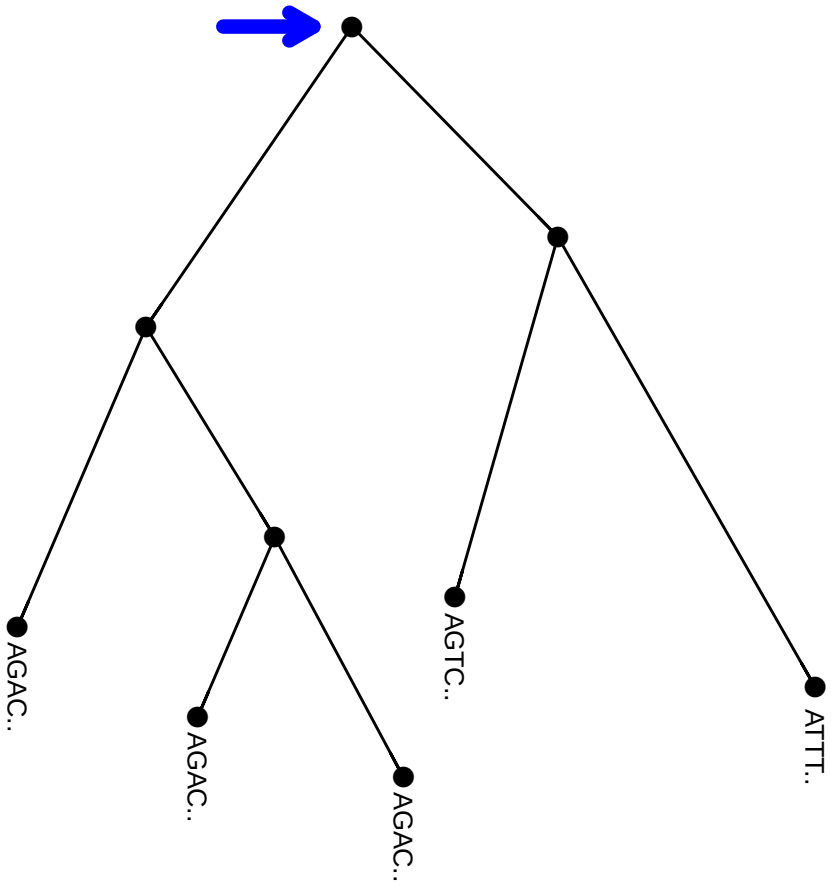
A  **CTCTAGCCTTTTTCCACT**

B  **TTCTAGCCTTTCTCTACT**

C  **CTCTAGCCTTTCTCTACT**

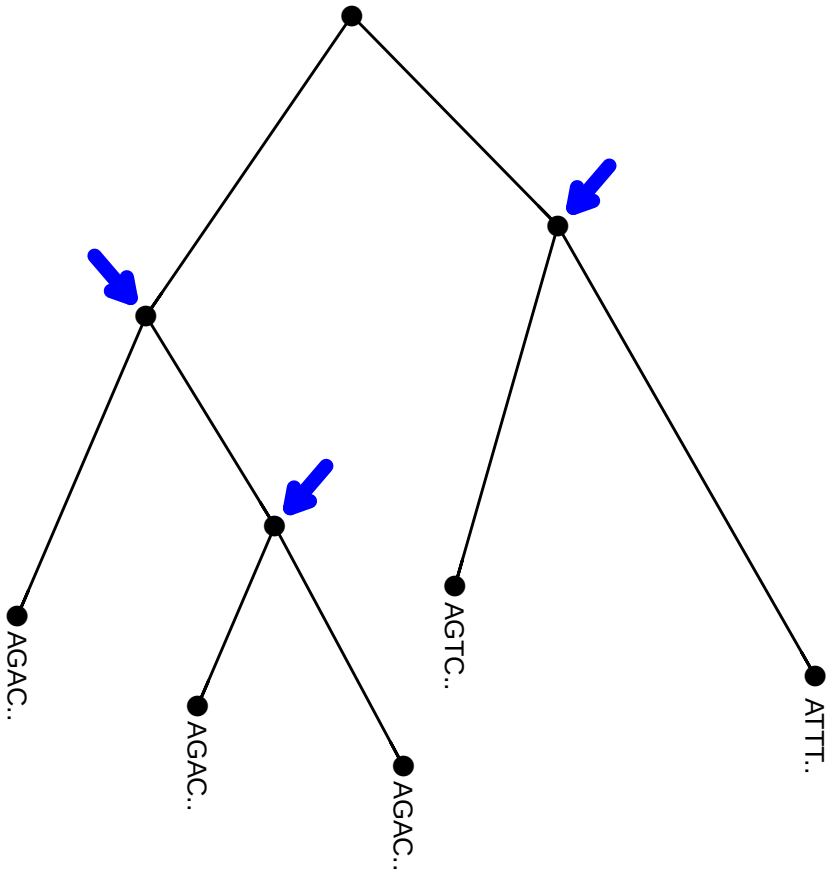
A rooted phylogenetic tree, t, and evolution

- Root: a common ancestor;



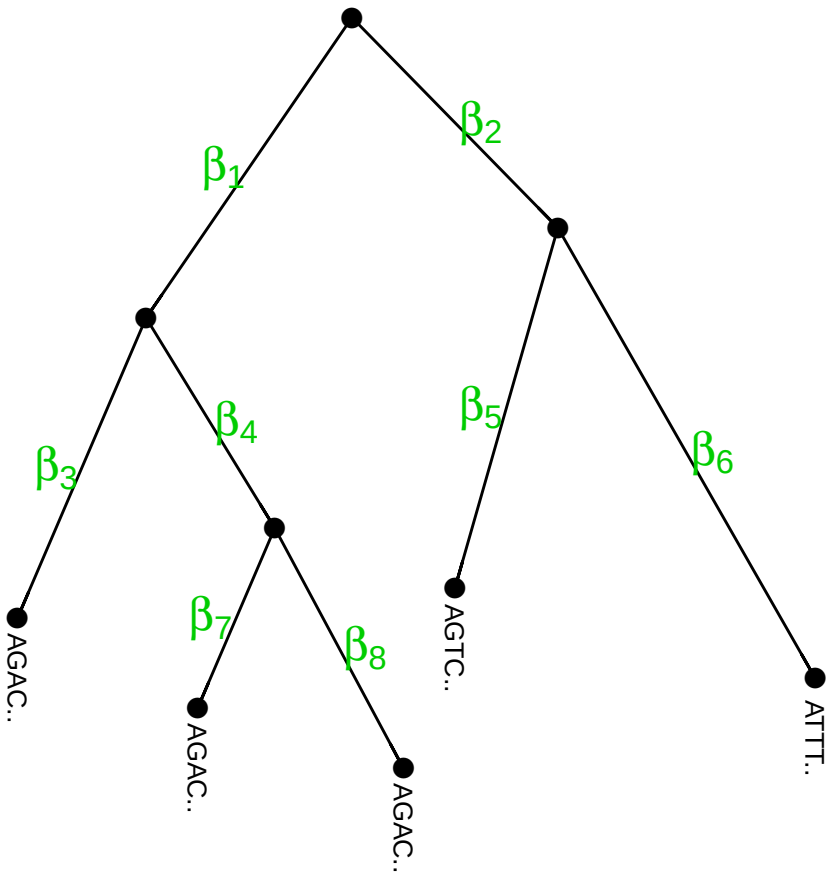
A rooted phylogenetic tree, t, and evolution

- Root: a common ancestor;
- Internal nodes



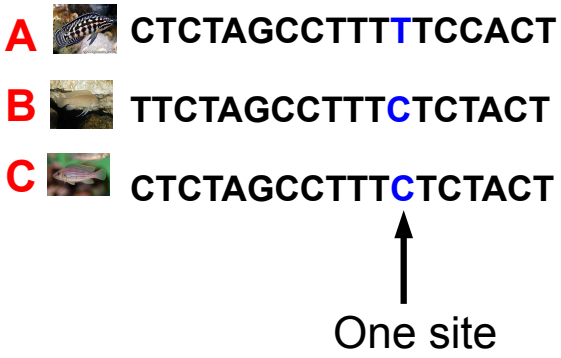
A rooted phylogenetic tree, t, and evolution

- Root: a common ancestor;
- Internal nodes
- Branch lengths
  - positive real numbers associated with each edge,
  - specifying the amount of evolution between nodes.



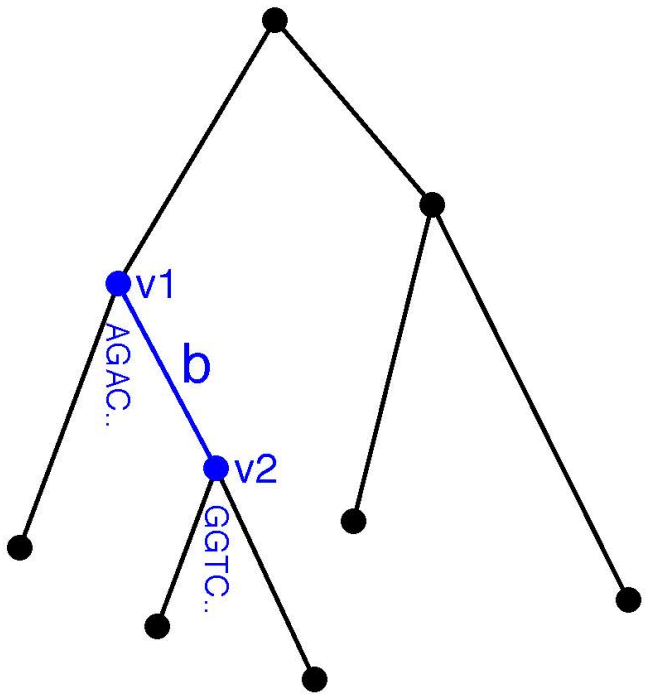
A likelihood model: $\mathbb{P}(\mathcal{Y}|t,\theta)$

- Assumption: site independence.



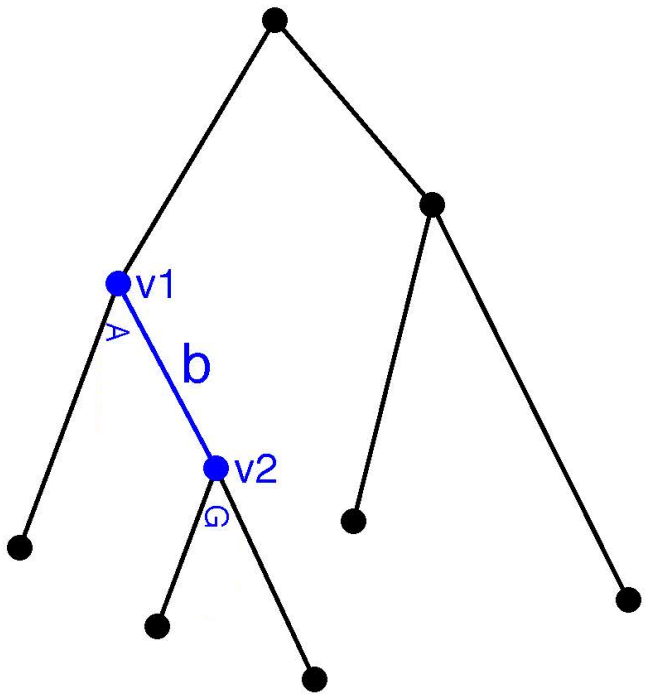
A likelihood model: $\mathbb{P}(\mathcal{Y}|t,\theta)$

- Assumption: site independence.
- Likelihood model on each site over one branch is a *Continuous Time Markov Chain* (CTMC): $\{Y_s : s \in [0,b]\}$
- The state space of the chain: $Y_s \in \{A,C,G,T\}$.

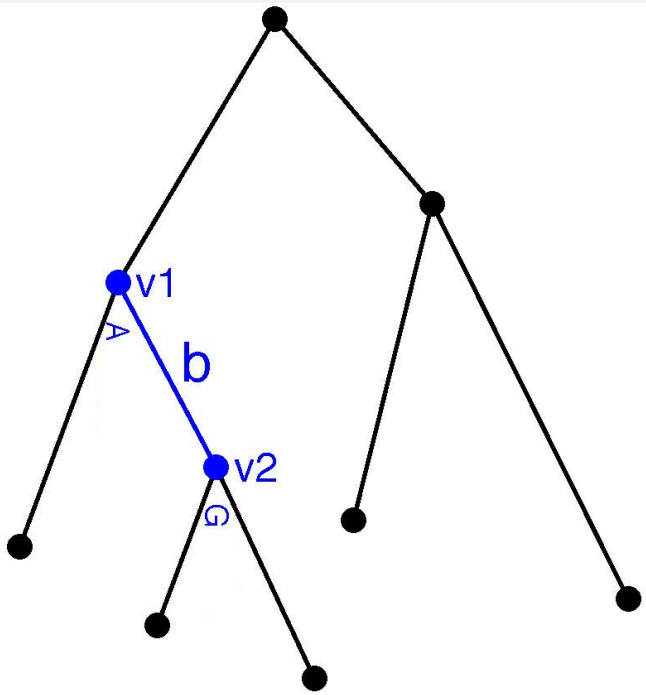


A likelihood model: $\mathbb{P}(\mathcal{Y}|t,\theta)$

- Assumption: site independence.
- Likelihood model on each site over one branch is a *Continuous Time Markov Chain* (CTMC): $\{Y_s : s \in [0,b]\}$
- The state space of the chain: $Y_s \in \{A, C, G, T\}$.



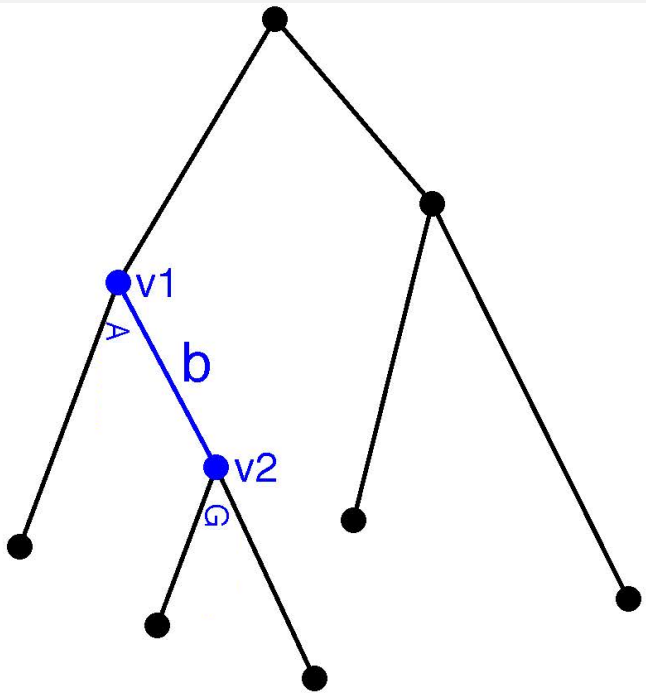
A likelihood model: $\mathbb{P}(\mathcal{Y}|t,\theta)$



- Assumption: site independence.
- Likelihood model on each site over one branch is a *Continuous Time Markov Chain* (CTMC): $\{Y_s : s \in [0,b]\}$
- The state space of the chain: $Y_s \in \{A, C, G, T\}$.
- The transition matrix: $P(b) = e^{Q_{4\times4}b}$ for the branch of length b.
- e.g. $P_{1,3}(b) = P(Y_b = G|Y_0 = A)$.

$$Q = \begin{bmatrix} - & \pi_C & \kappa\pi_G & \pi_T \\ \pi_A & - & \pi_G & \kappa\pi_T \\ \kappa\pi_A & \pi_C & - & \pi_T \\ \pi_A & \kappa\pi_C & \pi_G & - \end{bmatrix}$$

A likelihood model: $\mathbb{P}(\mathcal{Y}|t,\theta)$



- Assumption: site independence.
- Likelihood model on each site over one branch is a *Continuous Time Markov Chain* (CTMC): $\{Y_s : s \in [0,b]\}$
- The state space of the chain: $Y_s \in \{A, C, G, T\}$.
- The transition matrix: $P(b) = e^{Q_{4\times4}b}$ for the branch of length b.
- e.g. $P_{1,3}(b) = P(Y_b = G|Y_0 = A)$.
- Evolutionary parameters in CTMCs are denoted by θ.

$$Q = \begin{bmatrix} - & \pi_C & \kappa\pi_G & \pi_T \\ \pi_A & - & \pi_G & \kappa\pi_T \\ \kappa\pi_A & \pi_C & - & \pi_T \\ \pi_A & \kappa\pi_C & \pi_G & - \end{bmatrix}$$

Bayesian phylogenetics

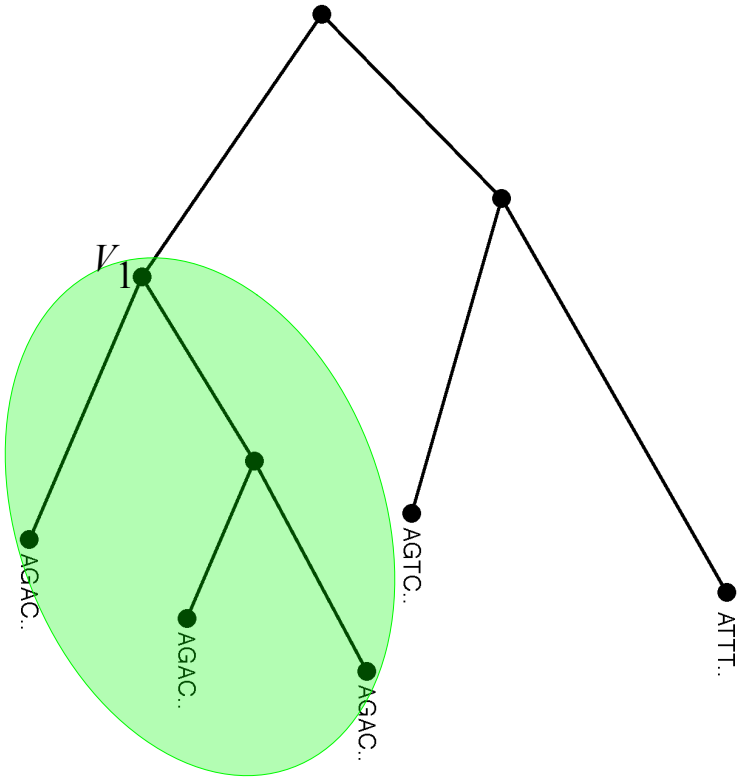
- Data: aligned sequences, denoted by $\mathcal{Y}$
- θ: evolutionary parameters
- t: a phylogenetic tree
- Posterior

$$\pi(\theta, t|\mathcal{Y}) = \frac{\mathbb{P}(\mathcal{Y}|t, \theta)p(t|\theta)p(\theta)}{\mathbb{P}(\mathcal{Y})}$$

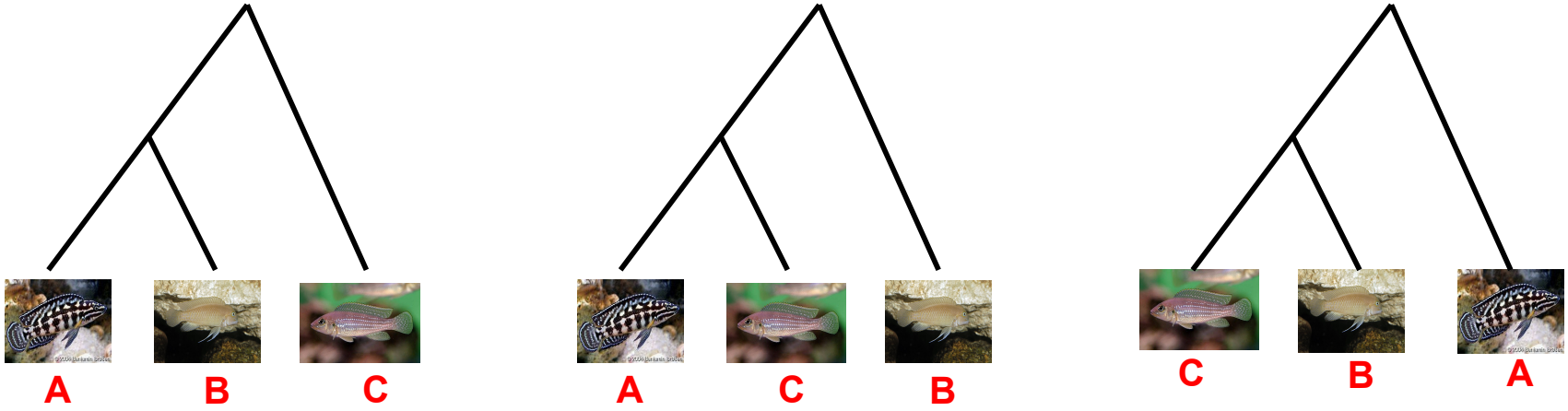
Posterior expectation of $\varphi(t)$: $\int \pi(dt)\varphi(t)$

An example of the function φ:

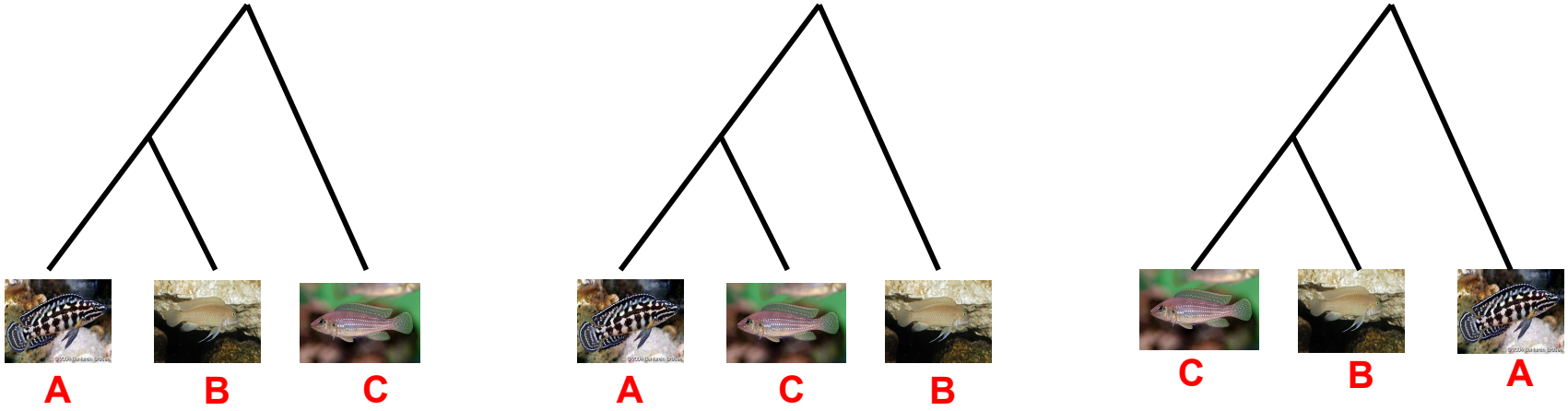
- $\varphi(t) = \mathbf{1}(c \in \text{clades}(t))$
- a **clade**: a group consisting of a species and all its descendants
- clades(t): all the clades of the tree t



Difficult inference problem over a huge tree space



Difficult inference problem over a huge tree space

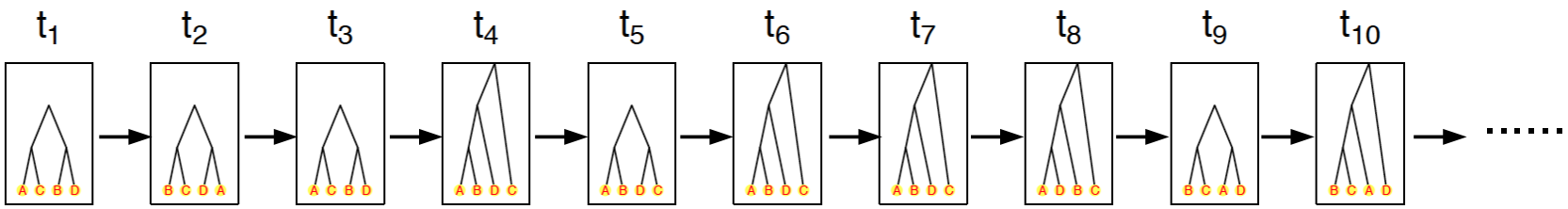


- Tree space for a phylogenetic tree

#Species	#Topologies
3	3
4	15
6	945
10	34459425

Standard Bayesian phylogenetics using MCMC

MCMC: obtain samples $t_k \sim \pi(\cdot|\mathcal{Y}), k = 1, \cdots, K$



$$\int \pi(dt)\varphi(t) \approx \frac{1}{K}\sum_{k=1}^{K}\varphi(t_k)$$

Problems with MCMC

- The Markov chain doesn't explore the tree space well
- Only small moves are allowed in each iteration
- Each step is expensive to compute
- MCMC does not scale to large datasets
  - a large number of taxa
  - large amount of data for each taxon

The Ultimate Goal in Phylogenetics

Infer phylogenetic trees accurately and efficiently

Develop new statistical evolutionary models

Computational algorithms for efficient analysis of large-scale datasets

The Ultimate Goal in Phylogenetics

Infer phylogenetic trees accurately and efficiently

Develop new statistical evolutionary models

- Current model: CTMC over characters for each site.
- Proposed model: a general string-valued CTMC for biological sequences.

Computational algorithms for efficient analysis of large-scale datasets

The Ultimate Goal in Phylogenetics

Infer phylogenetic trees accurately and efficiently

Develop new statistical evolutionary models

- Current model: CTMC over characters for each site.
- Proposed model: a general string-valued CTMC for biological sequences.

Computational algorithms for efficient analysis of large-scale datasets

- Current methods:
  - Standard MCMC
  - SMC for unrealistic phylogenetic trees (Teh et al. 2008; Bouchard-Côté et al. 2011) for fixed parameters θ

The Ultimate Goal in Phylogenetics

Infer phylogenetic trees accurately and efficiently

Develop new statistical evolutionary models

- Current model: CTMC over characters for each site.
- Proposed model: a general string-valued CTMC for biological sequences.

Computational algorithms for efficient analysis of large-scale datasets

- Current methods:
  - Standard MCMC
  - SMC for unrealistic phylogenetic trees (Teh et al. 2008; Bouchard-Côté et al. 2011) for fixed parameters θ
- Proposed methods:
  - An efficient SMC algorithm for **general** phylogenetic trees

The Ultimate Goal in Phylogenetics

Infer phylogenetic trees accurately and efficiently

Develop new statistical evolutionary models

- Current model: CTMC over characters for each site.
- Proposed model: a general string-valued CTMC for biological sequences.

Computational algorithms for efficient analysis of large-scale datasets

- Current methods:
  - Standard MCMC
  - SMC for unrealistic phylogenetic trees (Teh et al. 2008; Bouchard-Côté et al. 2011) for fixed parameters θ
- Proposed methods:
  - An efficient SMC algorithm for **general** phylogenetic trees
  - PMCMC for joint estimation of t and θ

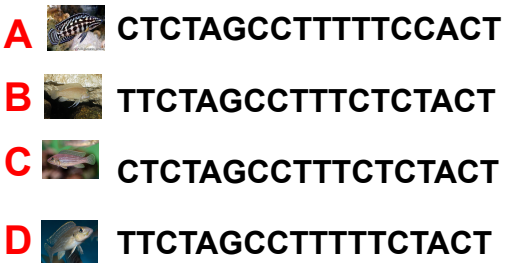
SMC algorithm for phylogenetic trees

- Target distribution: the posterior $\pi(t|\mathcal{Y}) \propto \gamma(t|\mathcal{Y}) = \mathbb{P}(\mathcal{Y}|t)p(t)$
- Interested in the posterior expectation of $\varphi(t)$: $\int \pi(dt)\varphi(t)$.

SMC algorithm for phylogenetic trees

- Target distribution: the posterior $\pi(t|\mathcal{Y}) \propto \gamma(t|\mathcal{Y}) = \mathbb{P}(\mathcal{Y}|t)p(t)$
- Interested in the posterior expectation of $\varphi(t)$: $\int \pi(dt)\varphi(t)$.

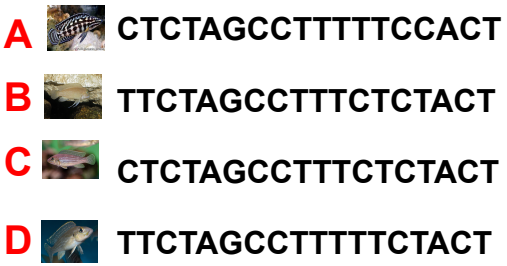
- Input $\mathcal{Y}$
  - Aligned biological sequences



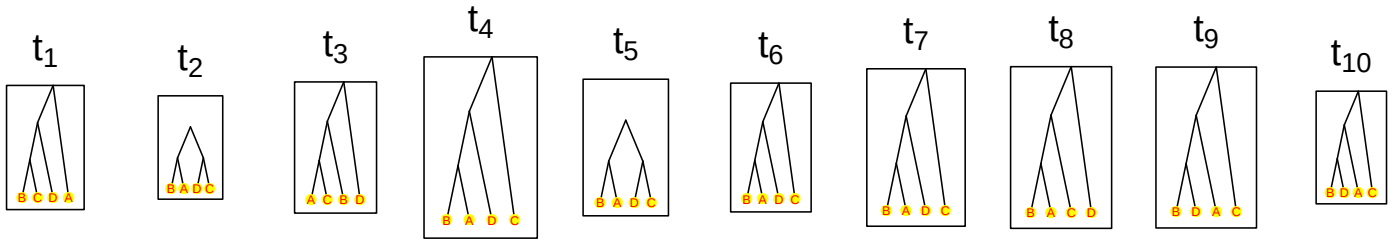
SMC algorithm for phylogenetic trees

- Target distribution: the posterior $\pi(t|\mathcal{Y}) \propto \gamma(t|\mathcal{Y}) = \mathbb{P}(\mathcal{Y}|t)p(t)$
- Interested in the posterior expectation of $\varphi(t)$: $\int \pi(dt)\varphi(t)$.

- Input $\mathcal{Y}$
  - Aligned biological sequences



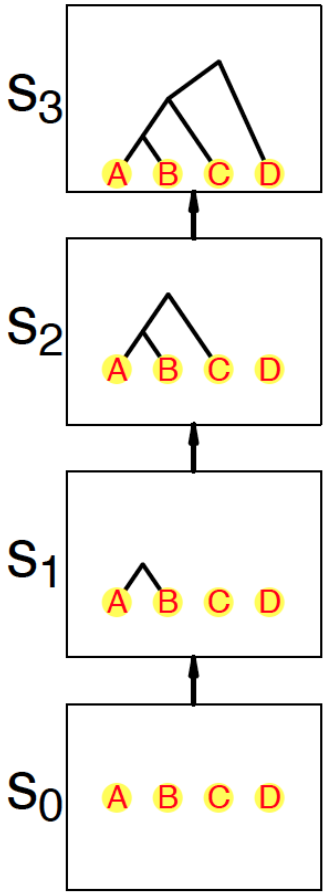
- Output:
  - weighted particles $\{(t_k, W_k)\}$ to approximate the posterior distribution over trees, $\hat{\pi}(t|\mathcal{Y})$



  - estimate of the marginal likelihood, $\hat{\mathbb{P}}(\mathcal{Y})$.

A sequence of partial states

- Using $\nu^+(s_0 \to t)$ is not efficient
- s_r: a partial state (forest) of $n - r$ subtrees (n is the number of species)
- A forward proposal $\nu^+(s_{r-1} \to s_r)$: randomly choose a pair of subtrees of s_{r-1} to merge.

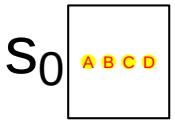


How to define distributions π_r over partial states s_r?

- $\pi_r(s_r|\mathcal{Y}) \propto \mathbb{P}(\mathcal{Y}|s_r)p(s_r)$
- $\mathbb{P}(\mathcal{Y}|s_r)$: the likelihood of the partial state s_r
  - we have likelihood model for trees
  - consider the trees in a forest to be independent
  - the product of the likelihood of the subtrees of s_r.
- π_r is represented by K weighted particles, $\{(s_{rk}, W_{rk})\}$

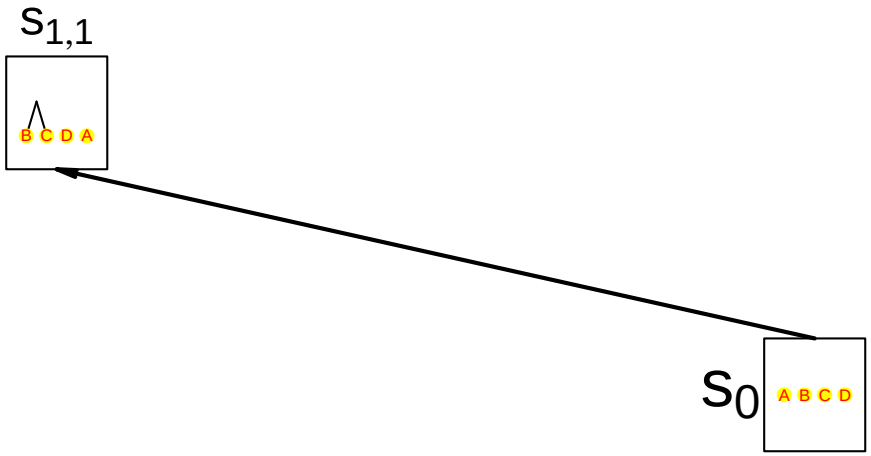
The initial partial state

- s_0: a forest in which each sequence is a trivial tree with a single leaf.



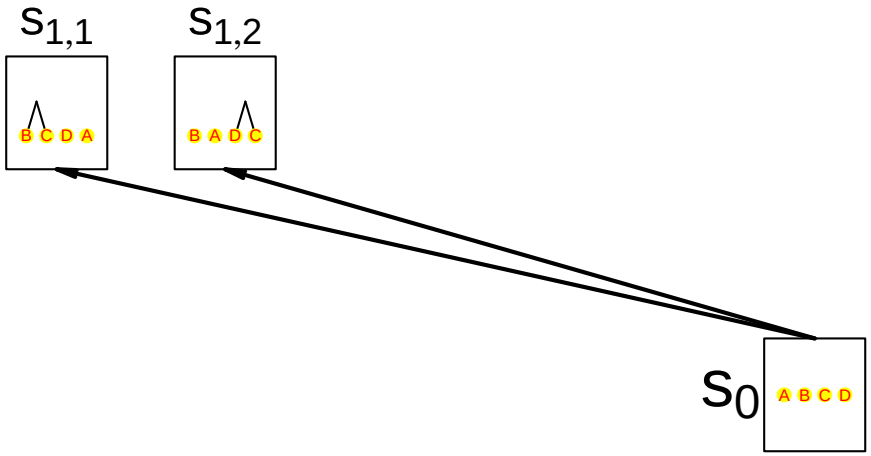
The first partial state s_1

- Generate particle s_{11} using $\nu^+(s_0 \to \cdot)$
- Randomly choose a pair of subtrees, species B and C, to merge.



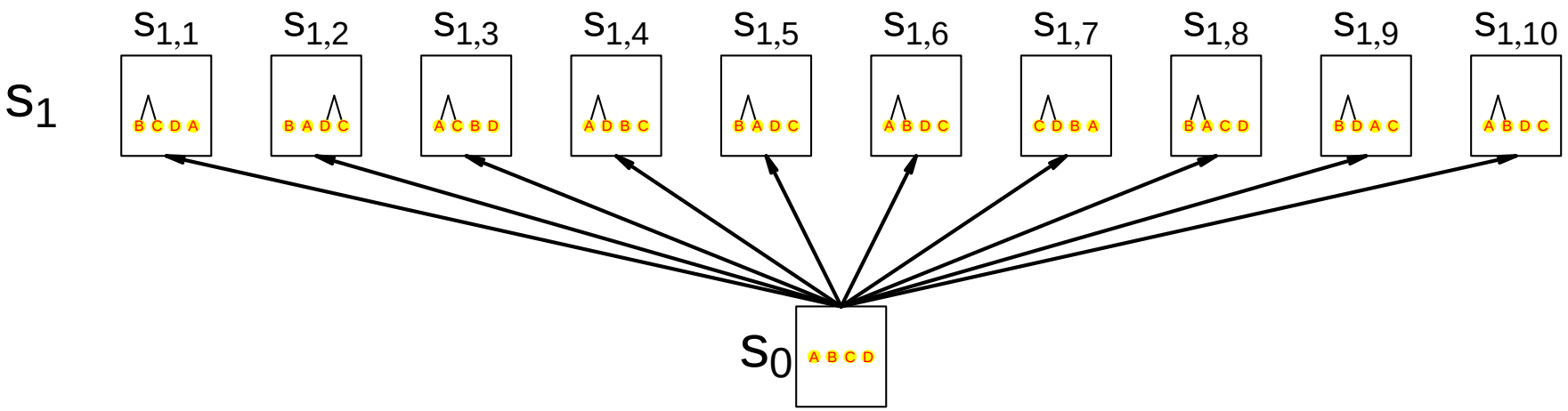
The first partial state s_1

- Generate particle s_{12} using $\nu^+(s_0 \to \cdot)$
- Randomly choose a pair of subtrees, species D and C, to merge.



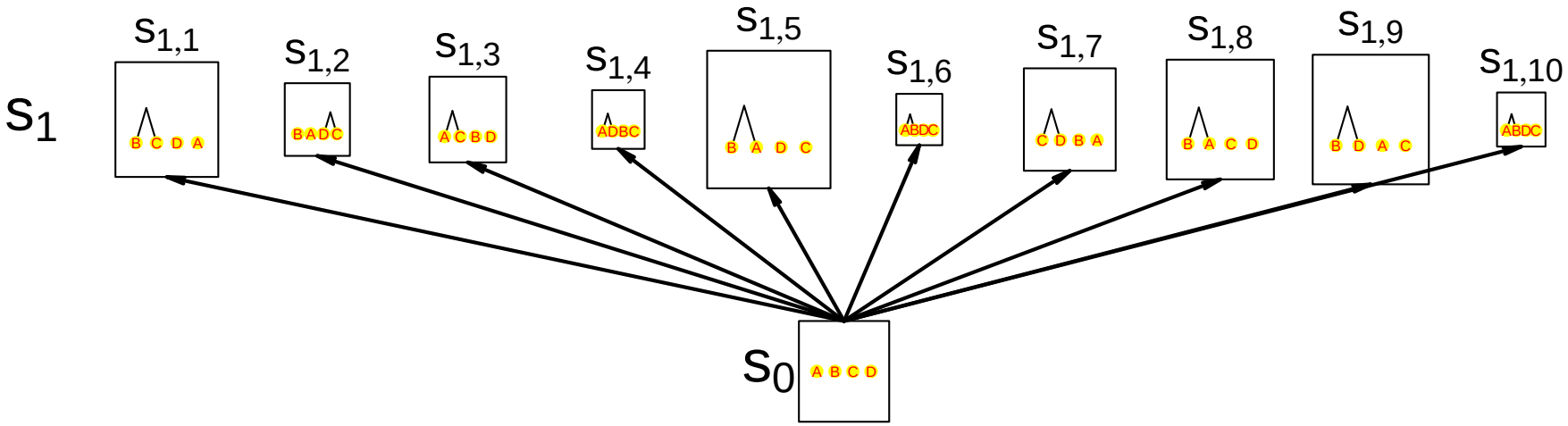
The first partial state s_1

- Generate K particles s_{1k} using $\nu^+(s_0 \to \cdot)$
- These particles cannot represent π_1 directly
- We need to compensate for the discrepancy between the distribution of interest and the proposed distribution.



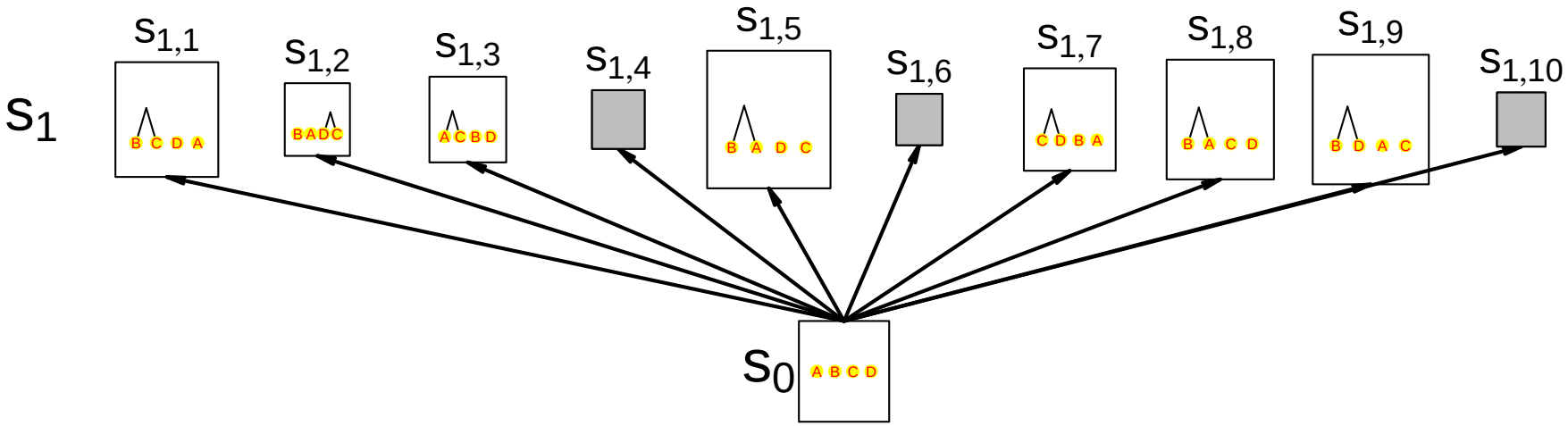
Update the weight of particles s_{1k}

- The rectangle size corresponds to the normalized particle weight $W_1(s_{1k})$.



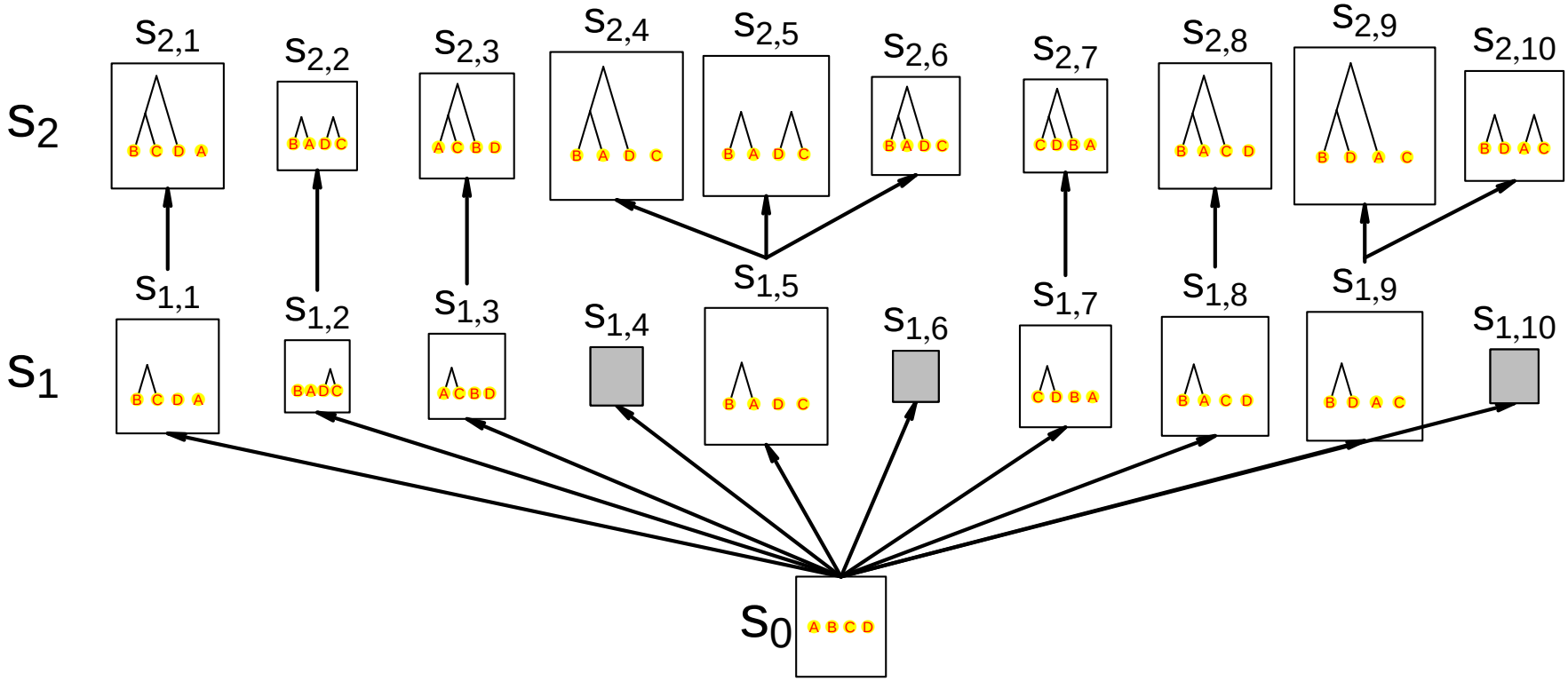
Resample s_{1k}

- Using a multinomial distribution
- Purpose: prune unpromising particles



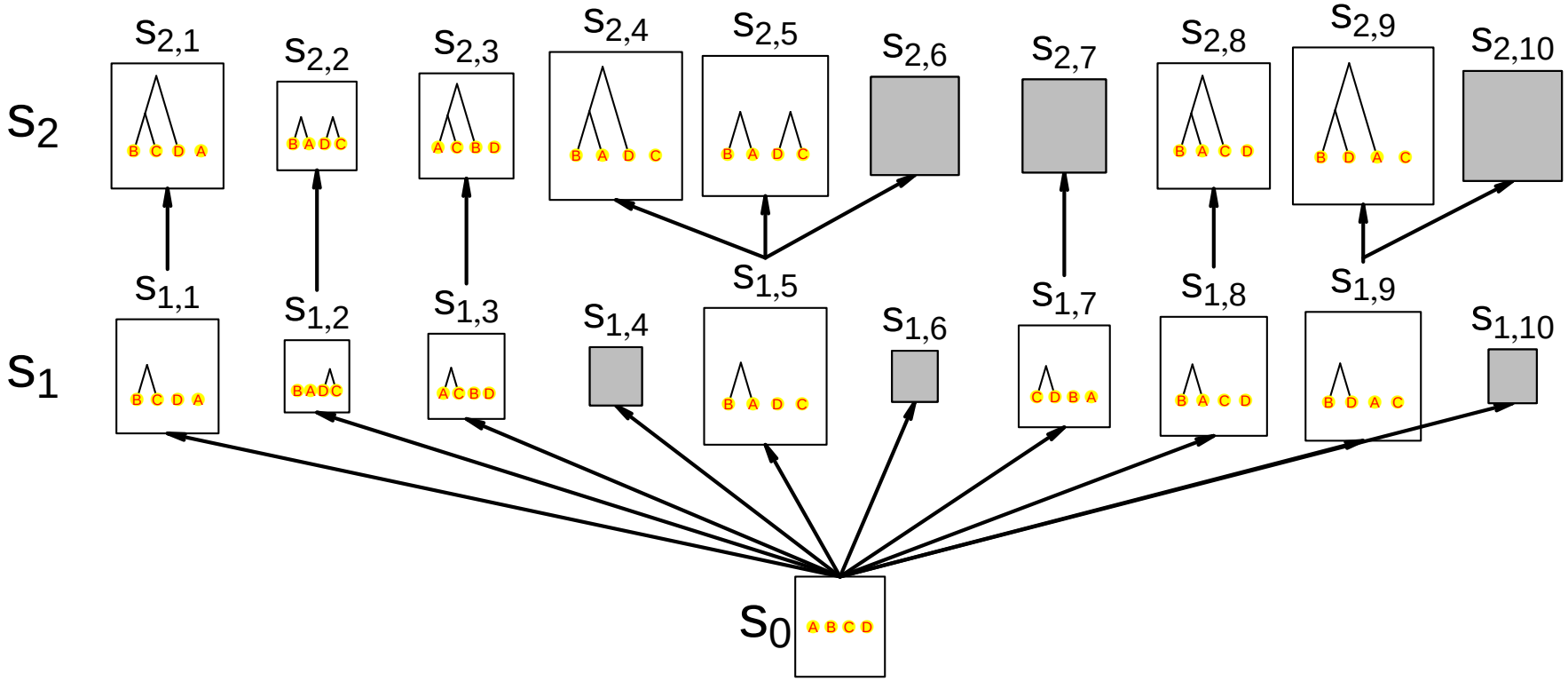
The second partial state s_2

- Generate the particles s_{2k} using $\nu^+(s_{1k} \to \cdot)$
- Update the weights of the particles $W_2(s_{2k})$

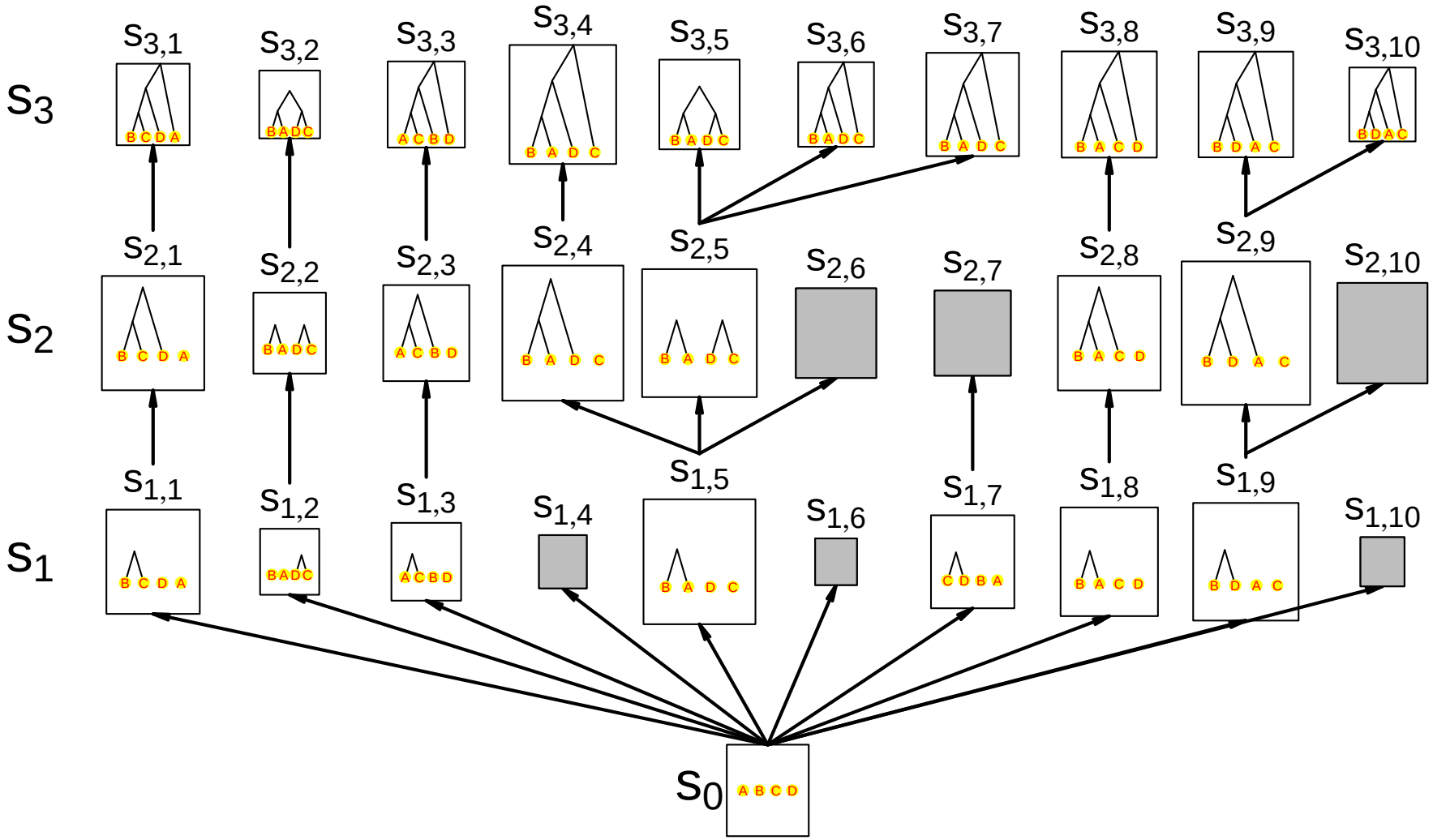


Resample s_{2k}

- Using a multinomial distribution.
- Purpose: prune unpromising particles



The final state (full tree)



The weight update in a standard SMC

$$w_r(s_r) = w_{r-1}(s_{r-1}) \cdot \frac{\gamma_r(s_r)}{\gamma_{r-1}(s_{r-1})} \frac{1}{\nu^+(s_{r-1} \to s_r)}$$

- $\gamma_r(s_r)$: unnormalized density of s_r
- $\gamma_{r-1}(s_{r-1})$: unnormalized density of s_{r-1}
- ν^+ : forward proposal

The weight update in a standard SMC

$$w_r(s_r) = w_{r-1}(s_{r-1}) \cdot \frac{\gamma_r(s_r)}{\gamma_{r-1}(s_{r-1})} \frac{1}{\nu^+(s_{r-1} \to s_r)}$$

- $\gamma_r(s_r)$: unnormalized density of s_r
- $\gamma_{r-1}(s_{r-1})$: unnormalized density of s_{r-1}
- ν^+ : forward proposal

This weight update of the standard SMC will lead to a biased estimate for general phylogenetic trees due to an over-counting problem.

The weight update in a standard SMC

$$w_r(s_r) = w_{r-1}(s_{r-1}) \cdot \frac{\gamma_r(s_r)}{\gamma_{r-1}(s_{r-1})} \frac{1}{\nu^+(s_{r-1} \to s_r)}$$

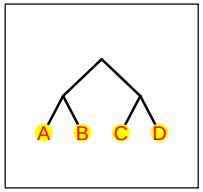
WRONG for general trees!

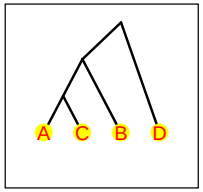
- $\gamma_r(s_r)$: unnormalized density of s_r
- $\gamma_{r-1}(s_{r-1})$: unnormalized density of s_{r-1}
- ν^+ : forward proposal

This weight update of the standard SMC will lead to a biased estimate for general phylogenetic trees due to an over-counting problem.

Over-counting some samples

Two trees (each 1/2)

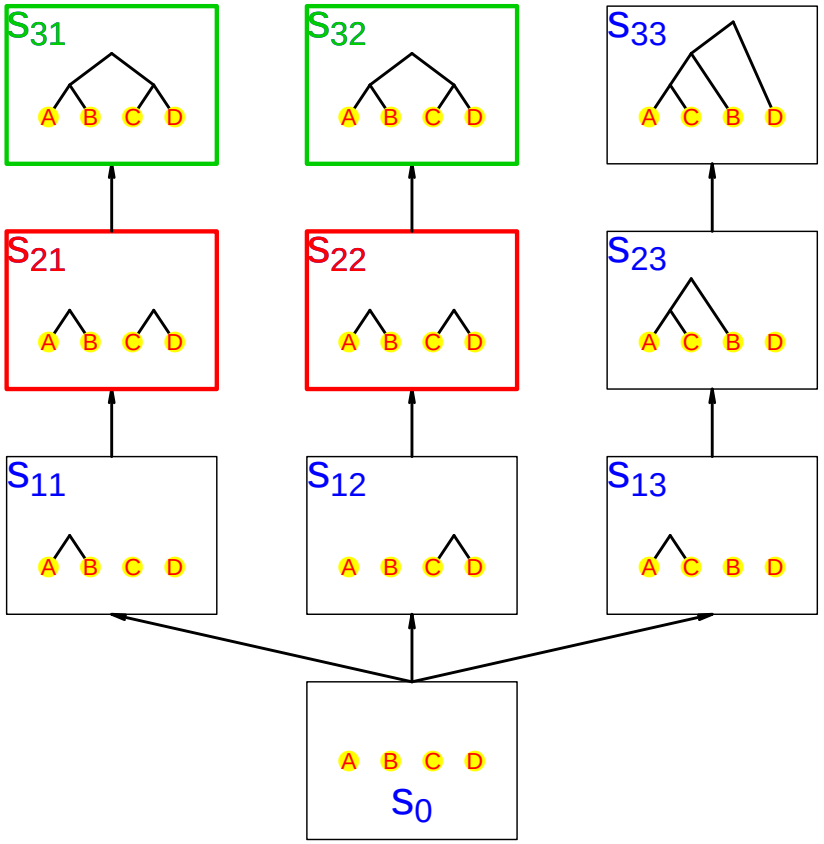




Over-counting some samples

Two trees (each 1/2)

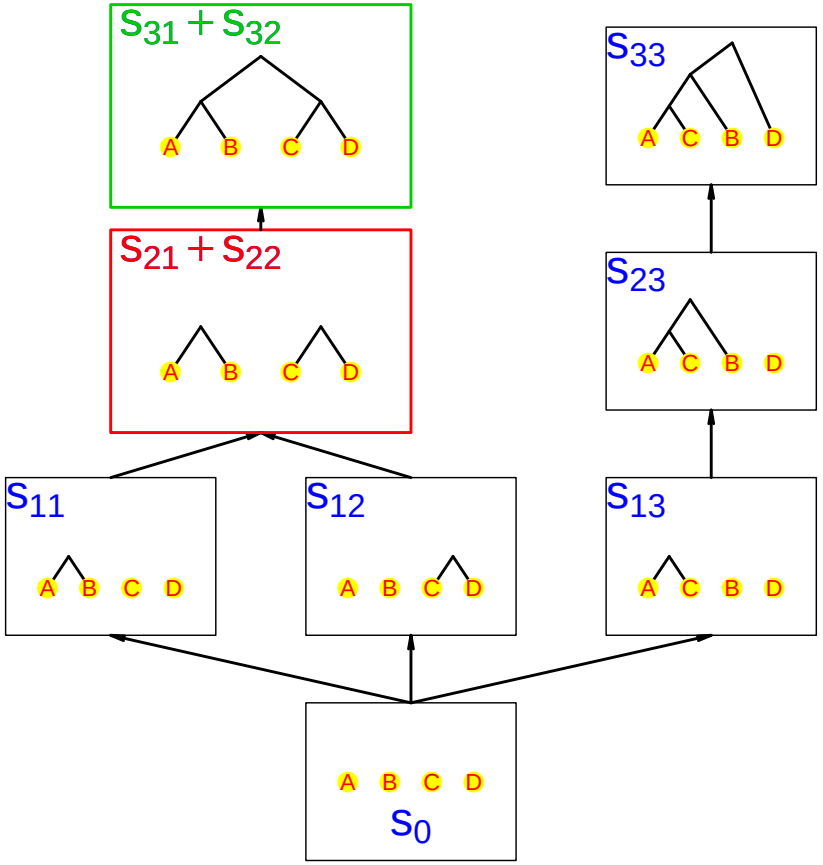
- Two copies of the same partial state: s_{21}, s_{22}
- Two copies of the same full tree: s_{31}, s_{32}



Over-counting some samples

Two trees (each 1/2)

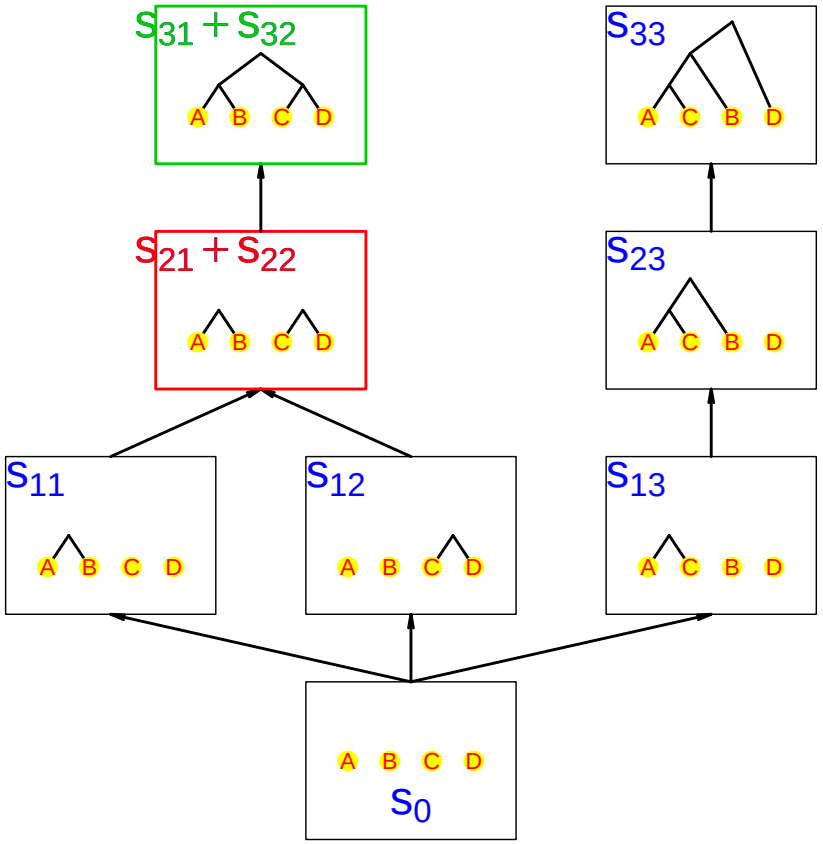
- Two copies of the same partial state: s_{21}, s_{22}
- Two copies of the same full tree: s_{31}, s_{32}
- Their weights are doubled.
- This will cause biased estimates.



Over-counting some samples

Two trees (each 1/2)

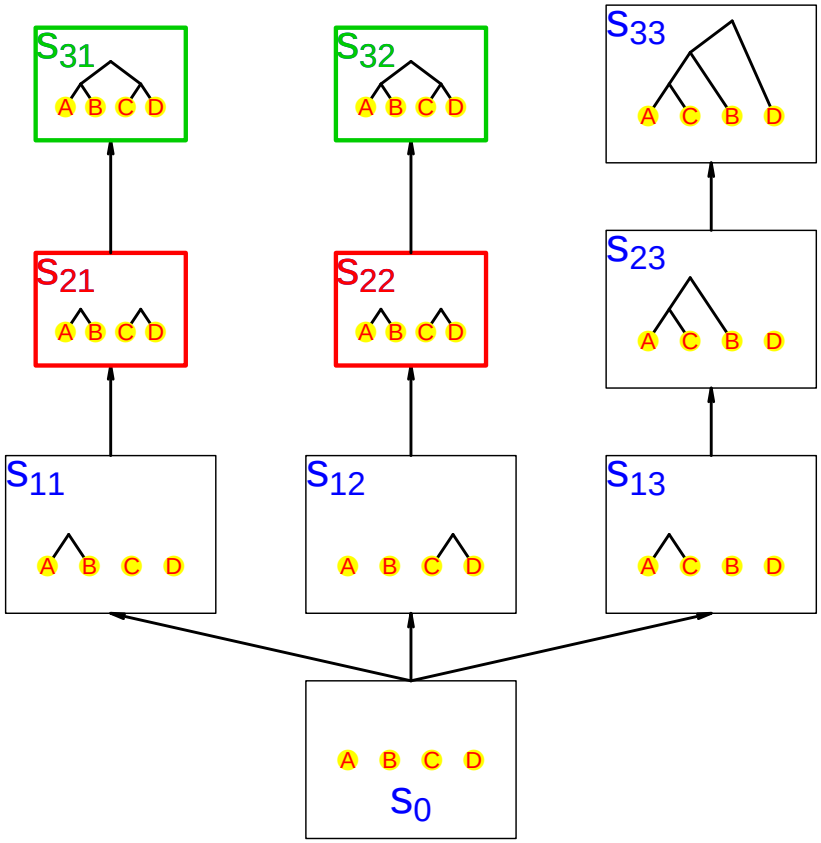
- Two copies of the same partial state: s_{21}, s_{22}
- Two copies of the same full tree: s_{31}, s_{32}
- Their weights are doubled.
- This will cause biased estimates.
- Need to downweight the over-counted partial states.



Over-counting some samples

Two trees (each 1/2)

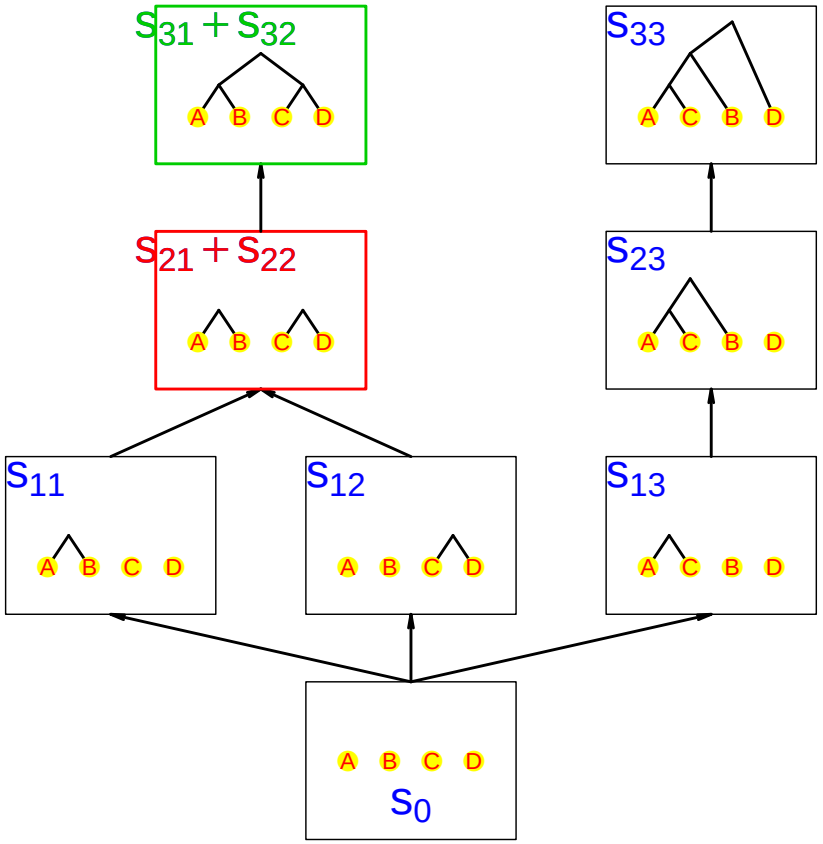
- Two copies of the same partial state: s_{21}, s_{22}
- Two copies of the same full tree: s_{31}, s_{32}
- Their weights are doubled.
- This will cause biased estimates.
- Need to downweight the over-counted partial states.



Over-counting some samples

Two trees (each 1/2)

- Two copies of the same partial state: s_{21}, s_{22}
- Two copies of the same full tree: s_{31}, s_{32}
- Their weights are doubled.
- This will cause biased estimates.
- Need to downweight the over-counted partial states.



Our correct weight update (in the CSMC algorithm)

$$w_r(s_r) = w_{r-1}(s_{r-1}) \cdot \frac{\gamma_r(s_r)}{\gamma_{r-1}(s_{r-1})} \frac{\nu^-(s_r \to s_{r-1})}{\nu^+(s_{r-1} \to s_r)}$$

The backward proposal ν^- is

- based on a graded partially ordered set (poset) on an extended combinatorial space
- 1: if there is only one way from s_{r-1} to s_r
- between 0 and 1 if there are multiple ways from s_{r-1} to s_r
  - to downweight the over-counted partial states.

Convergence results

Under weak conditions, for any bounded real-valued function $\varphi : \mathcal{S}_r \to \mathbb{R}$,

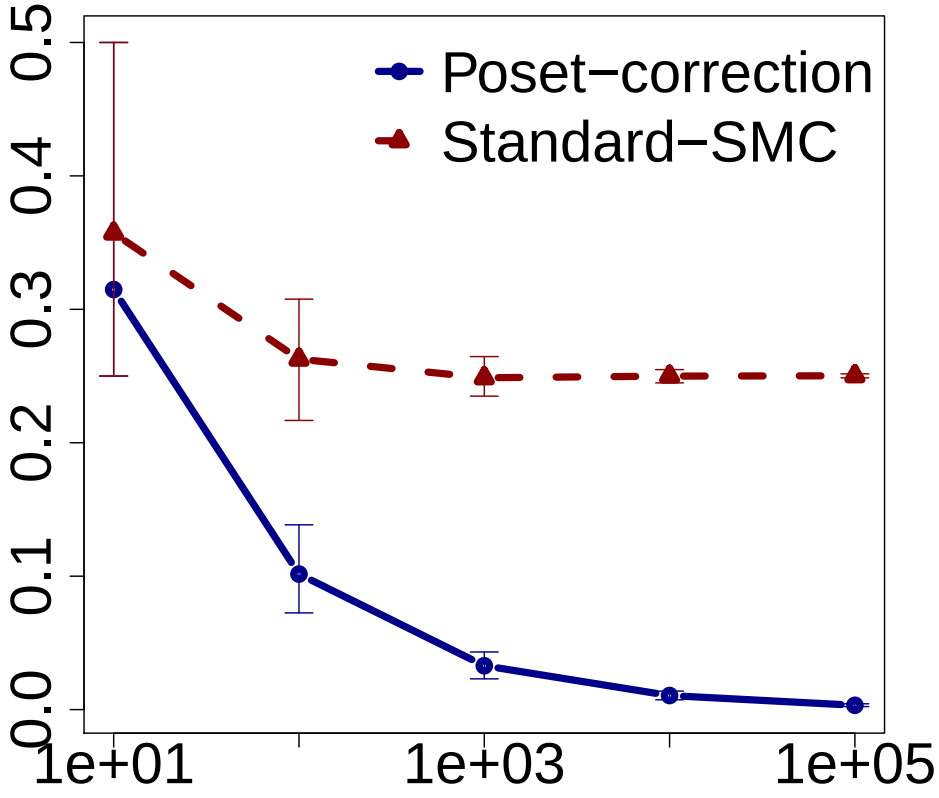
Strong Law of Large Numbers (SLLN)

$$\lim_{K\to\infty}\left(\sum_{k=1}^{K} W_{rk}\varphi(s_{rk}) - \int \pi_r(ds_r)\varphi(s_r)\right) \xrightarrow{a.s.} 0,$$

K: the number of weighted particles.

Illustration of convergence: simulation

y-axis: Total variation distance of $\hat{\pi}$ to π
x-axis: K (# particles)



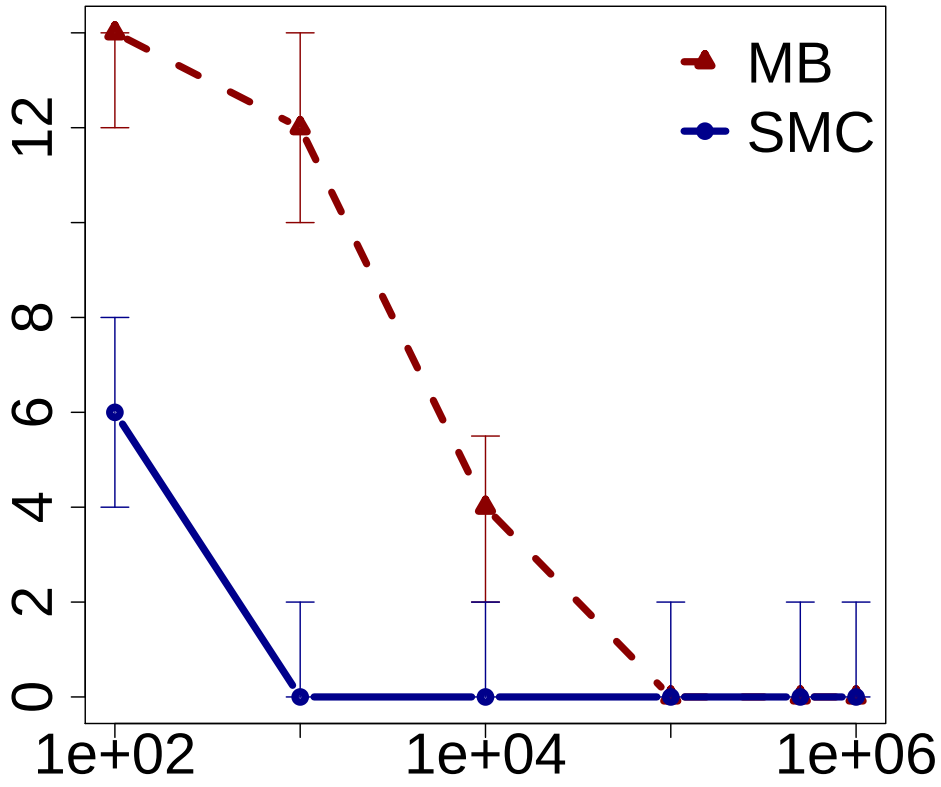
Experiment on tree inference

y-axis: Partition metric
x-axis: time (in log scale)

- # leaves: 10
- # sites: 1000
- # datasets: 1000

Computationally faster

100×: 2 orders of magnitude



Particle MCMC

Inferring both the tree and the evolutionary parameter jointly

$$\pi(\theta, t|\mathcal{Y})$$

Particle MCMC (Andrieu et al. 2010)

- Each MCMC iteration uses our proposed CSMC algorithm to approximate the posterior distribution of the phylogenetic tree
- Particle marginal Metropolis-Hastings (PMMH)
- Particle Independent Metropolis-Hastings (PIMH)
- The Particle Gibbs sampler (PGS)
  - requires a special SMC algorithm, conditional SMC.

Particle MCMC

Advantage

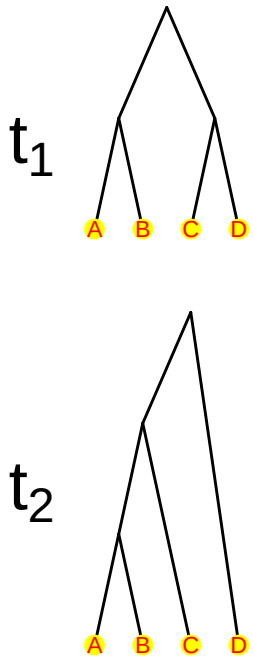
Bolder and more efficient move to update t.

Convergence result

These algorithms converges to the true posterior. (Andrieu et al. 2010)

Cartoon: problem with standard MCMC

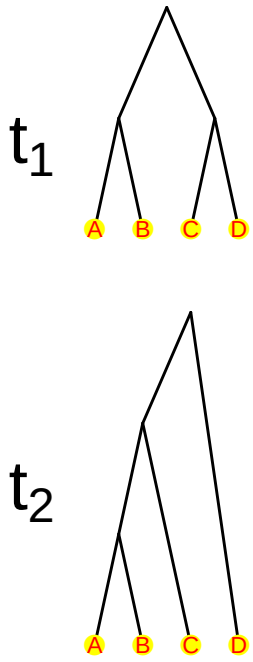
- Assume θ can only take two values: θ_1, θ_2
- Two possible trees: t_1, t_2
- At each iteration of MCMC, the chain is at one of 4 states: (θ_1, t_1), (θ_1, t_2), (θ_2, t_1), (θ_2, t_2)
- The square is a joint distribution
- A good Markov chain should move quickly among the states with high probability mass



	θ_1	θ_2
t_1	0.1	0.5
t_2	0.3	0.1

Cartoon: problem with standard MCMC

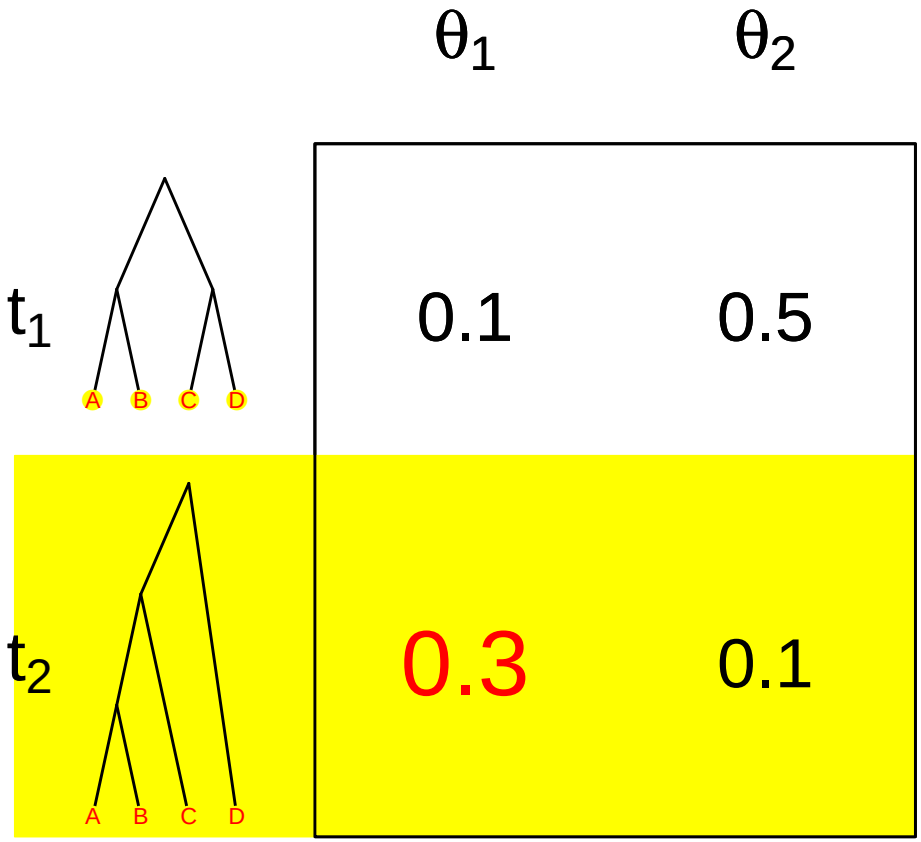
- Assume θ can only take two values: θ_1, θ_2
- Two possible trees: t_1, t_2
- At each iteration of MCMC, the chain is at one of 4 states: (θ_1, t_1), (θ_1, t_2), (θ_2, t_1), (θ_2, t_2)
- The square is a joint distribution
- A good Markov chain should move quickly among the states with high probability mass



	θ_1	θ_2
t_1	0.1	0.5
t_2	0.3	0.1

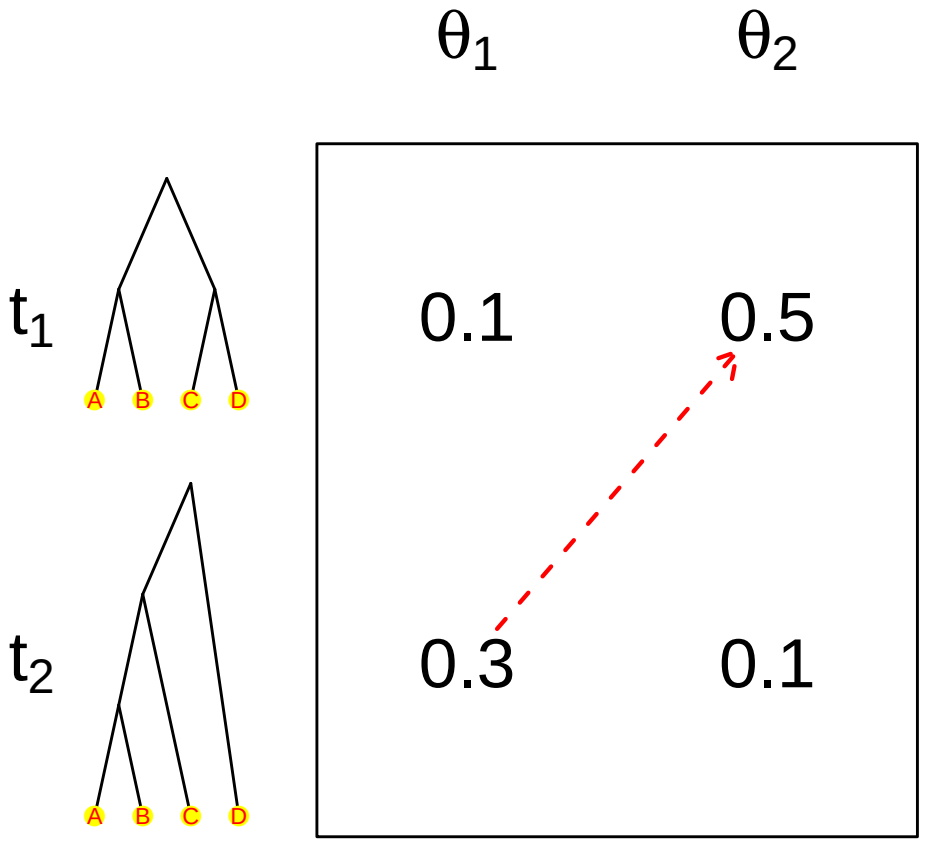
Cartoon: problem with standard MCMC

- Assume θ can only take two values: θ_1, θ_2
- Two possible trees: t_1, t_2
- At each iteration of MCMC, the chain is at one of 4 states: (θ_1, t_1), (θ_1, t_2), (θ_2, t_1), (θ_2, t_2)
- The square is a joint distribution
- A good Markov chain should move quickly among the states with high probability mass

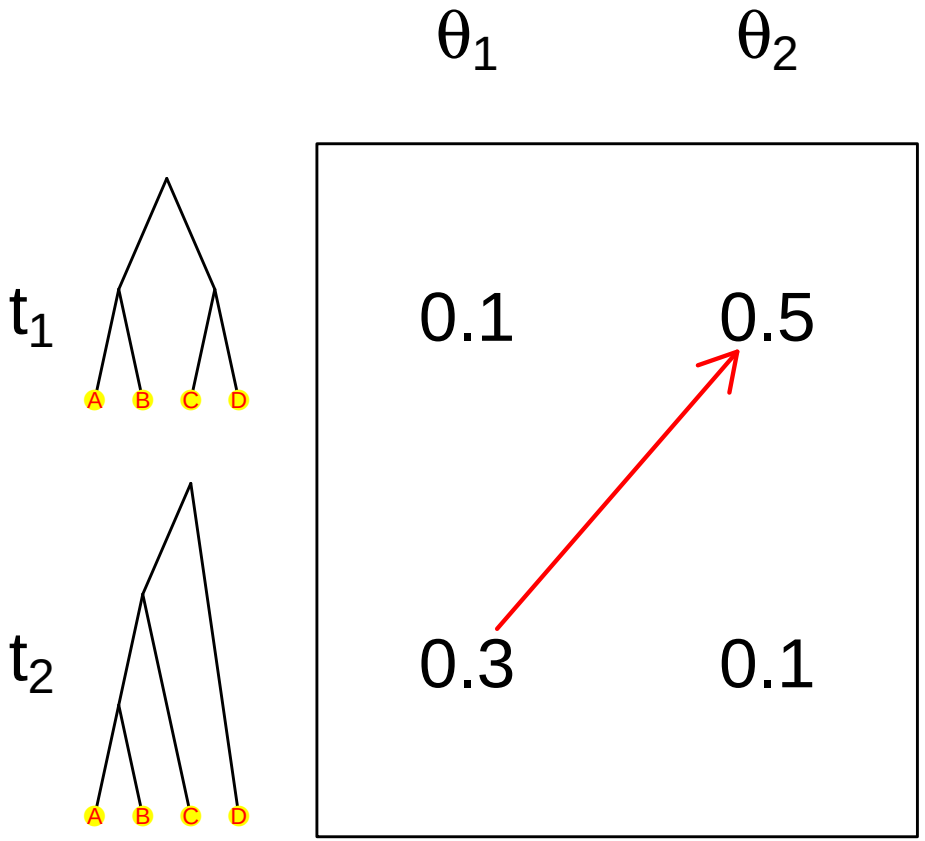


Cartoon: problem with standard MCMC

- Assume θ can only take two values: θ_1, θ_2
- Two possible trees: t_1, t_2
- At each iteration of MCMC, the chain is at one of 4 states: (θ_1, t_1), (θ_1, t_2), (θ_2, t_1), (θ_2, t_2)
- The square is a joint distribution
- A good Markov chain should move quickly among the states with high probability mass



Advantage of using particle MCMC



Particle marginal Metropolis-Hastings (PMMH)

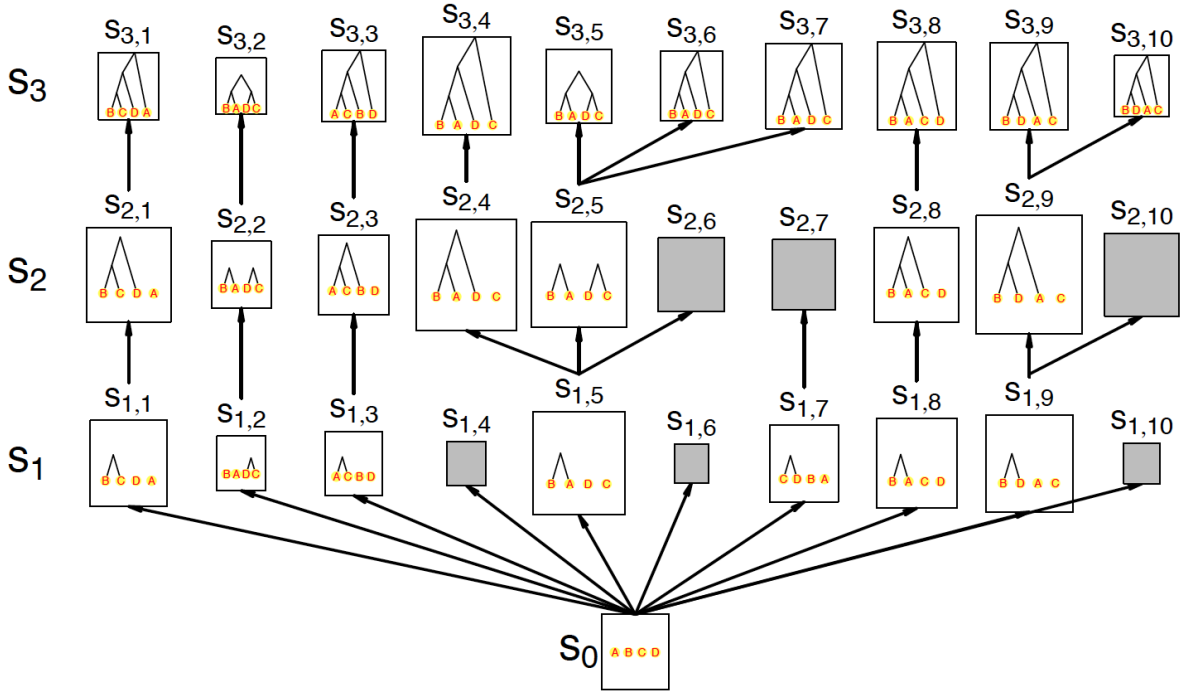
Each iteration of PMMH

1. sample $\theta^* \sim q(\theta \to \cdot)$,

Particle marginal Metropolis-Hastings (PMMH)

Each iteration of PMMH

1. sample $\theta^* \sim q(\theta \to \cdot)$,
2. run our SMC algorithm targeting $\pi_{\theta^*}(t|\mathcal{Y})$, sample $t^* \sim \hat{\pi}_{\theta^*}(\cdot|\mathcal{Y})$, and $\hat{\mathbb{P}}_{\theta^*}(\mathcal{Y})$ is the marginal likelihood obtained from SMC.



Particle marginal Metropolis-Hastings (PMMH)

Each iteration of PMMH

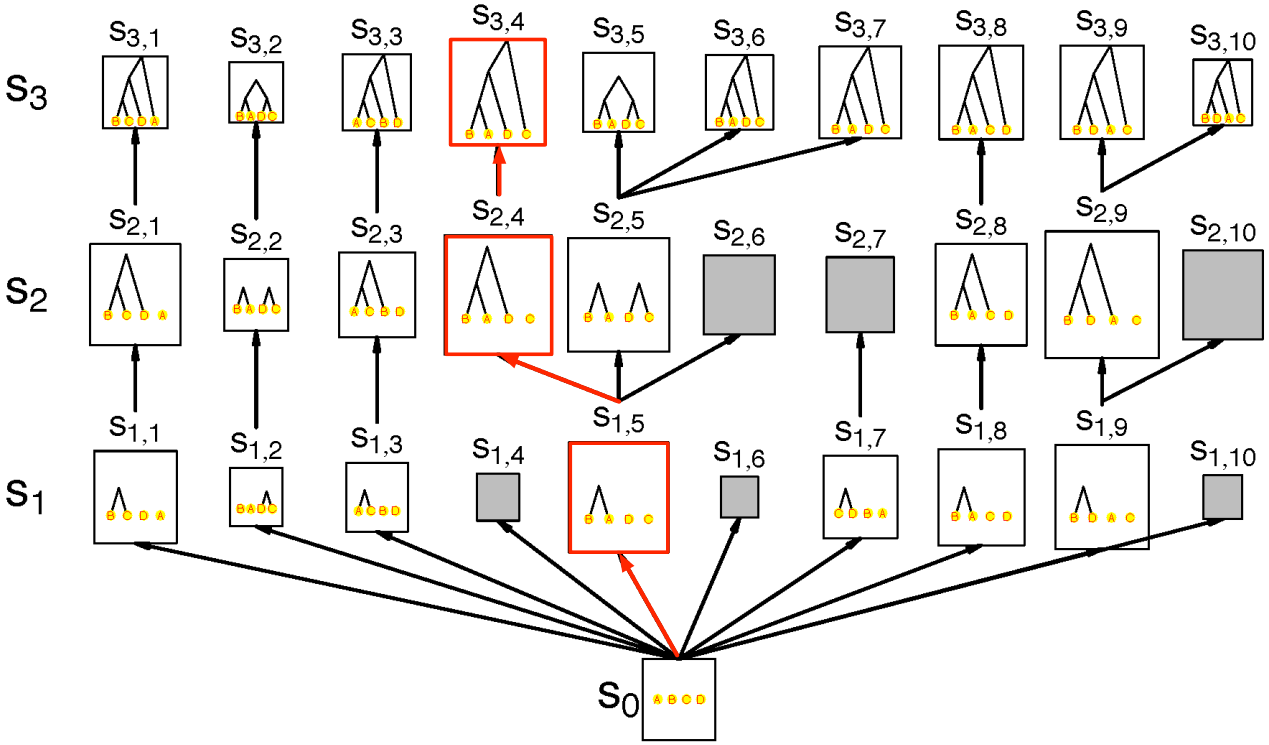
1. sample $\theta^* \sim q(\theta \to \cdot)$,
2. run our SMC algorithm targeting $\pi_{\theta^*}(t|\mathcal{Y})$, sample $t^* \sim \hat{\pi}_{\theta^*}(\cdot|\mathcal{Y})$, and $\hat{\mathbb{P}}_{\theta^*}(\mathcal{Y})$ is the marginal likelihood obtained from SMC.
3. Accept θ^* and t^* with the probability

$$\min\left(1, \frac{\hat{\mathbb{P}}_{\theta^*}(\mathcal{Y})p(\theta^*)}{\hat{\mathbb{P}}_{\theta}(\mathcal{Y})p(\theta)} \frac{q\{\theta^* \to \theta\}}{q\{\theta \to \theta^*\}}\right).$$

The Particle Gibbs sampler (PGS)

For each iteration

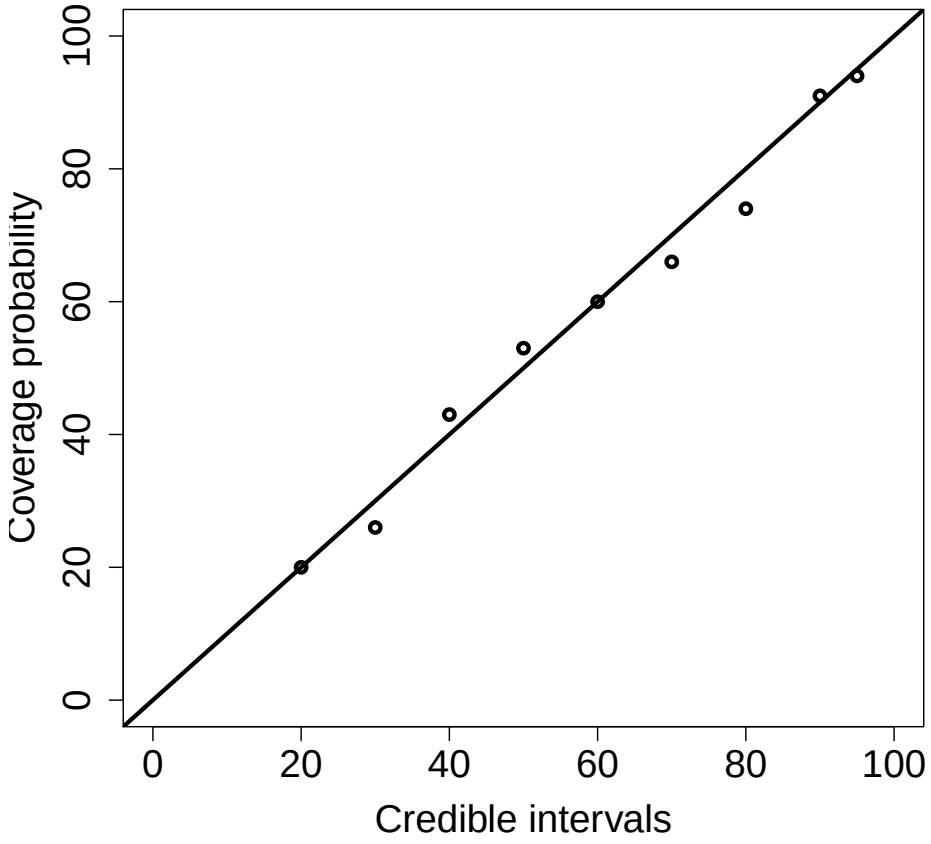
- Sample $\theta^* \sim p(\cdot|t)$
- Run the conditional CSMC algorithm targeting $\pi_{\theta^*}(t|\mathcal{Y})$ conditional on t and its ancestral lineage.
- Sample $t^* \sim \hat{\pi}_{\theta^*}(\cdot|\mathcal{Y})$.



Estimation of the parameters with PMMH

- True value: $\theta = 2$
- Using 100 datasets
- Averaged estimate: 1.99
- Standard deviation: 0.25

y-axis: Coverage probability
x-axis: Credible intervals

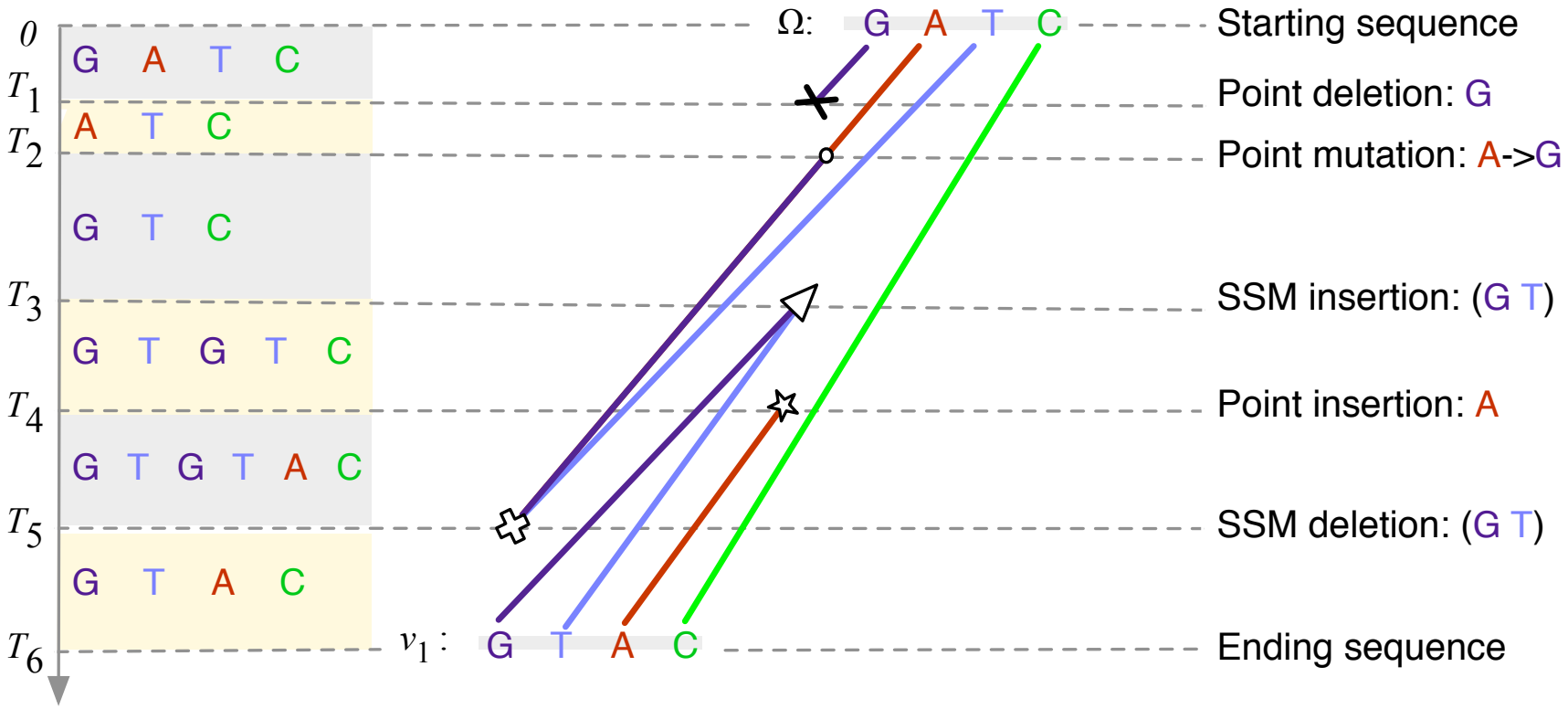


Ongoing and Future Work

- Harnessing non-Local evolutionary events for tree inference
  - Slipped strand mispairing (SSM)
- Joint estimation of Multiple Sequence Alignment (MSA) and phylogeny
- Inferring large scale trees on Graphics Processing Units (GPUs)

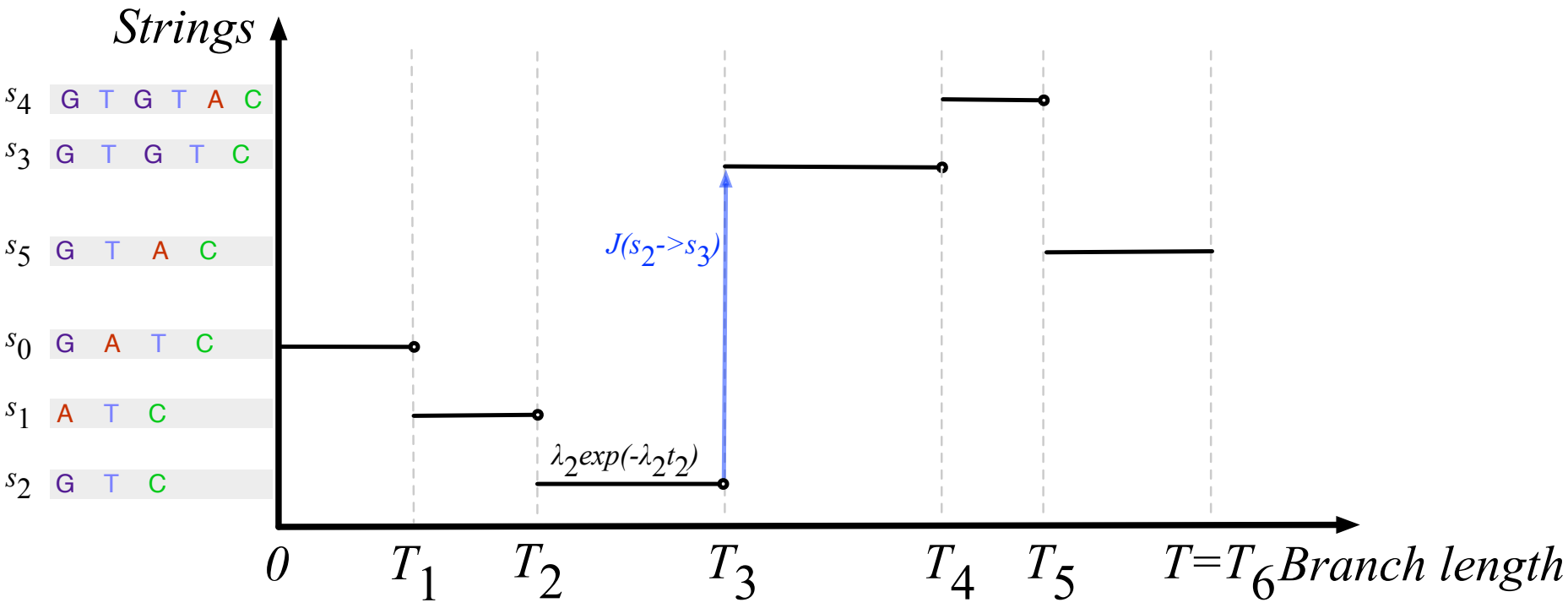
An Example of Evolutionary Events

Slipped Strand Mispairing (SSM) ⇒ long indels that depend on their contexts



String-valued Continuous Time Markov Chain (CTMC)

- This process is parametrized by the rate of departing from s, $\lambda(s)$, and the jumping distribution, $J(s \to \cdot)$.
- Waiting time at s: $t \sim Exp(\lambda(s))$; $\lambda_i = \lambda(s_i)$.



Summary

- A combinatorial SMC method
- Applicable to Bayesian inference in combinatorial spaces
- Converges to the true posterior asymptotically
- Computationally fast

Summary

- A combinatorial SMC method
- Particle MCMC

- Using the proposed SMC within MCMC iterations
- The Markov chain can explore the combinatorial space efficiently
- Accurate estimate of the parameters

Summary

- A combinatorial SMC method
- Particle MCMC
- Future work
- Harnessing non-Local evolutionary events for tree inference
- Joint estimation of MSA and phylogeny
- Inferring large scale trees on GPUs

Acknowledgements

Co-supervisors
Dr. Alexandre Bouchard-Côté
Dr. Arnaud Doucet

Thank you!

The Natural Sciences and Engineering Research Council of Canada (**NSERC**).

Bibliography

Andrieu, C., A. Doucet, and R. Holenstein (2010). Particle Markov chain Monte Carlo methods. *J. R. Statist. Soc. B 72*(3), 269–342.

Bouchard-Côté, A., S. Sankararaman, and M. I. Jordan (2011). Phylogenetic inference via sequential Monte Carlo. *Systematic Biology*.

Teh, Y. W., H. Daumé III, and D. M. Roy (2008). Bayesian agglomerative clustering with coalescents. In *Advances in Neural Information Processing Systems (NIPS)*.