Variational Bayesian Phylogenetic Inference

Cheng Zhang

joint work with Frederick Matsen

Fred Hutchinson Cancer Research Center, Seattle WA

Jun 10, 2019

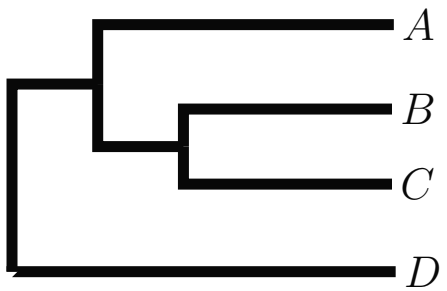
Introduction

Phylogenetic Inference

The goal of **phylogenetic inference** is to reconstruct the evolution history (e.g., *phylogenetic trees*) from molecular sequence data (e.g., DNA, RNA or protein sequences)

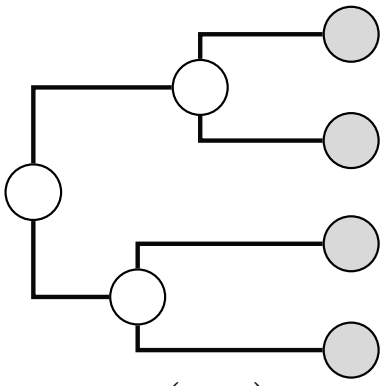
Taxa	Characters
Species A	ATGAACAT
Species B	ATGCACAC
Species C	ATGCATAT
Species D	ATGCATGC

Molecular Sequence Data



Phylogenetic Tree

Likelihood and Posterior



$(\tau, \boldsymbol{q})$

A T G A A C $\cdots$

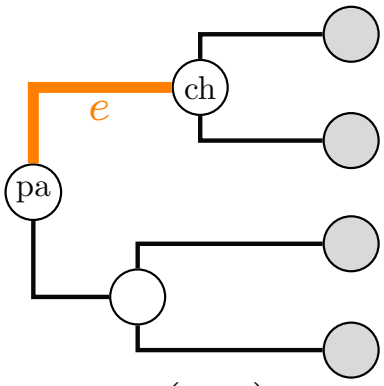
A T G C A C $\cdots$

A T G C A T $\cdots$

A T G C A T $\cdots$

$y_1\, y_2\, y_3\, y_4\, y_5\, y_6$

Likelihood and Posterior



$(\tau, \boldsymbol{q})$

A T G A A C $\cdots$

A T G C A C $\cdots$

A T G C A T $\cdots$

A T G C A T $\cdots$

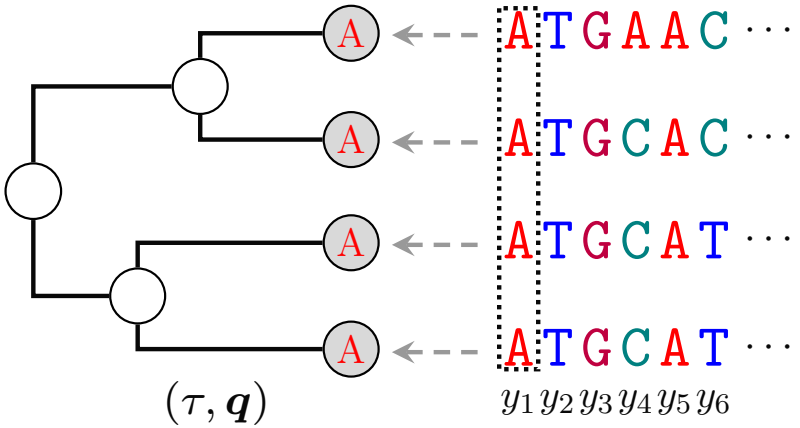
$y_1\, y_2\, y_3\, y_4\, y_5\, y_6$

Evolution model:

$$p(\text{ch}|\text{pa}, q_e)$$

q_e: amount of evolution on e.

Likelihood and Posterior



Evolution model:

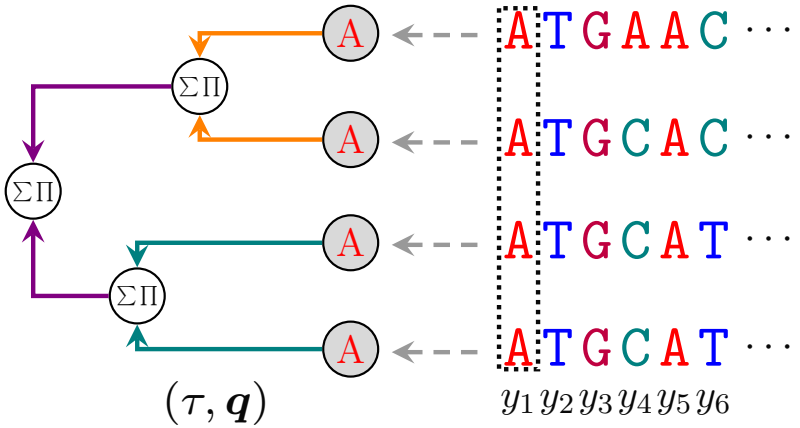
$$p(\text{ch}|\text{pa}, q_e)$$

q_e: amount of evolution on e.

Likelihood

$$p(\boldsymbol{Y}|\tau, \boldsymbol{q}) = \eta(a_\rho^i) \prod_{(u,v)\in E(\tau)} P_{a_u^i a_v^i}(q_{uv})$$

Likelihood and Posterior



Evolution model:

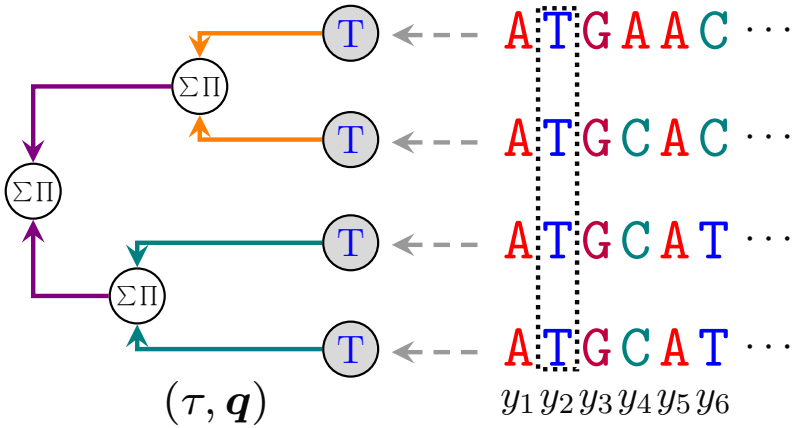
$$p(\text{ch}|\text{pa}, q_e)$$

q_e: amount of evolution on e.

Likelihood

$$p(\boldsymbol{Y}|\tau, \boldsymbol{q}) = \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v)\in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

Likelihood and Posterior



Evolution model:

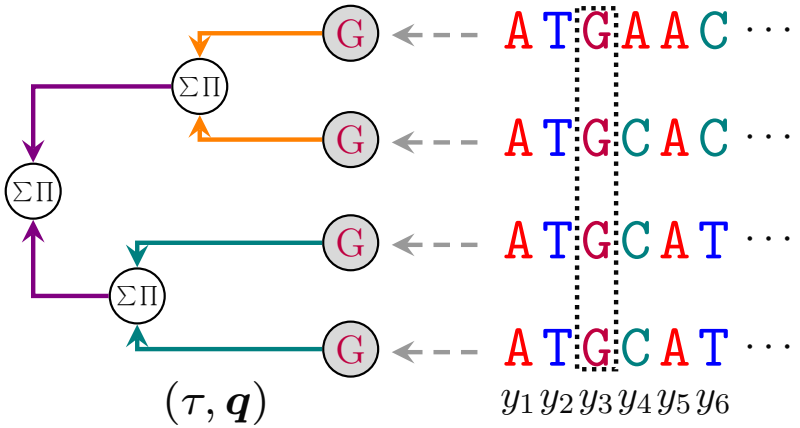
$$p(\text{ch}|\text{pa}, q_e)$$

q_e: amount of evolution on e.

Likelihood

$$p(\boldsymbol{Y}|\tau, \boldsymbol{q}) = \prod_{i=1}^{M} \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v)\in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

Likelihood and Posterior



Evolution model:

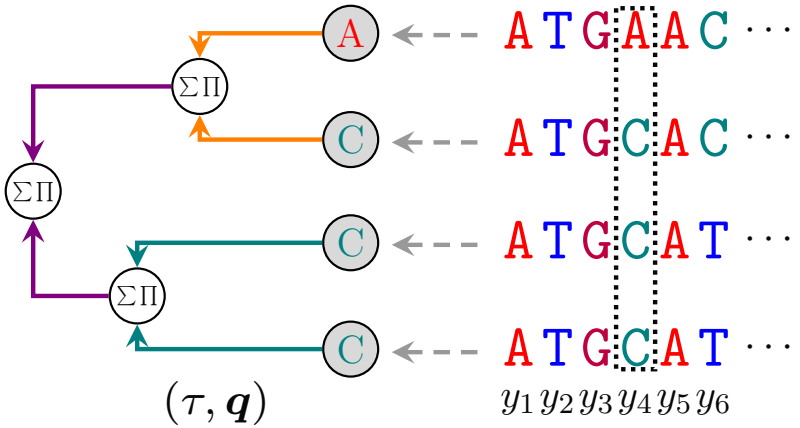
$$p(\text{ch}|\text{pa}, q_e)$$

q_e: amount of evolution on e.

Likelihood

$$p(\boldsymbol{Y}|\tau, \boldsymbol{q}) = \prod_{i=1}^{M} \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v)\in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

Likelihood and Posterior



Evolution model:

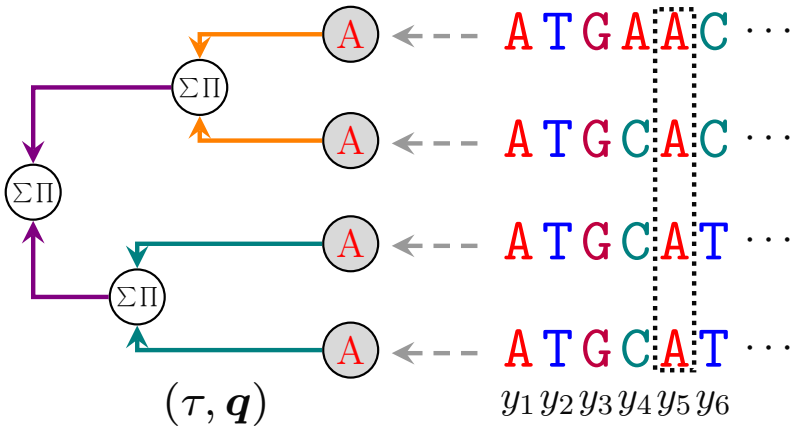
$$p(\text{ch}|\text{pa}, q_e)$$

q_e: amount of evolution on e.

Likelihood

$$p(\boldsymbol{Y}|\tau, \boldsymbol{q}) = \prod_{i=1}^{M} \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v)\in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

Likelihood and Posterior



Evolution model:

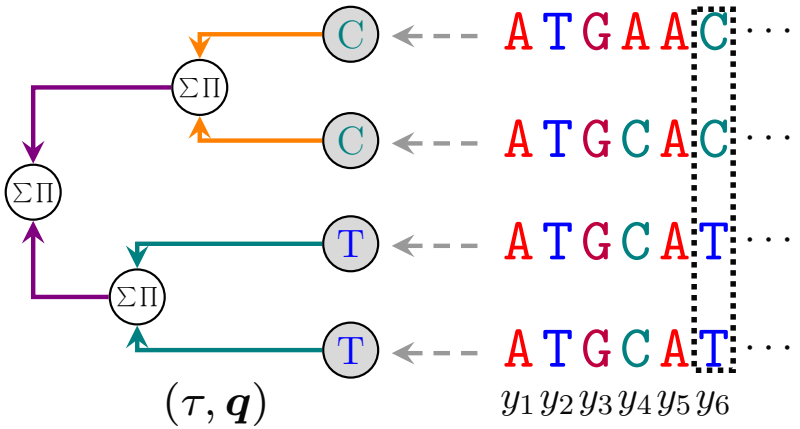
$$p(\text{ch}|\text{pa}, q_e)$$

q_e: amount of evolution on e.

Likelihood

$$p(\boldsymbol{Y}|\tau, \boldsymbol{q}) = \prod_{i=1}^{M} \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v)\in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

Likelihood and Posterior



Evolution model:

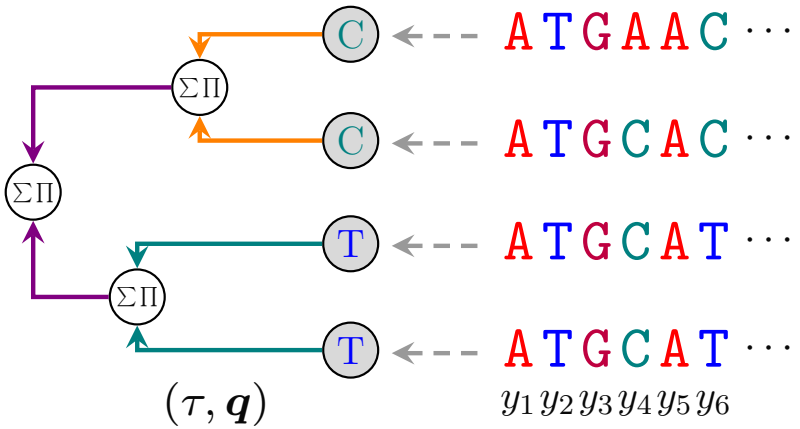
$$p(\text{ch}|\text{pa}, q_e)$$

q_e: amount of evolution on e.

Likelihood

$$p(\boldsymbol{Y}|\tau, \boldsymbol{q}) = \prod_{i=1}^{M} \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v)\in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

Likelihood and Posterior



Evolution model:

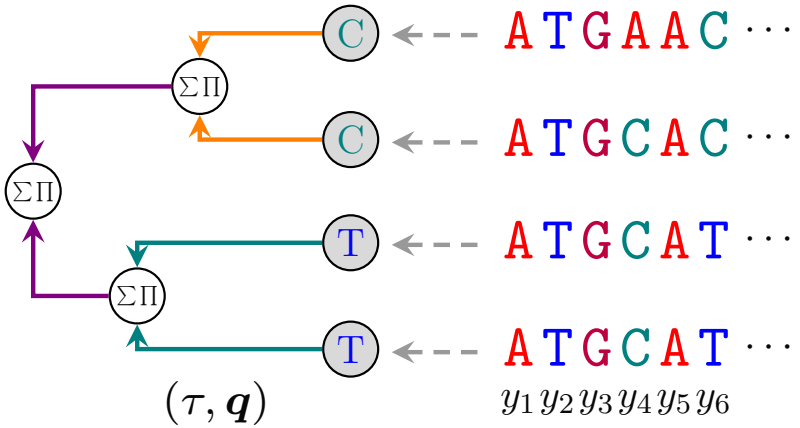
$$p(\mathrm{ch}|\mathrm{pa}, q_e)$$

q_e: amount of evolution on e.

Likelihood

$$p(\boldsymbol{Y}|\tau, \boldsymbol{q}) = \prod_{i=1}^{M} \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v)\in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

Likelihood and Posterior



Evolution model:

$$p(\text{ch}|\text{pa}, q_e)$$

q_e: amount of evolution on e.

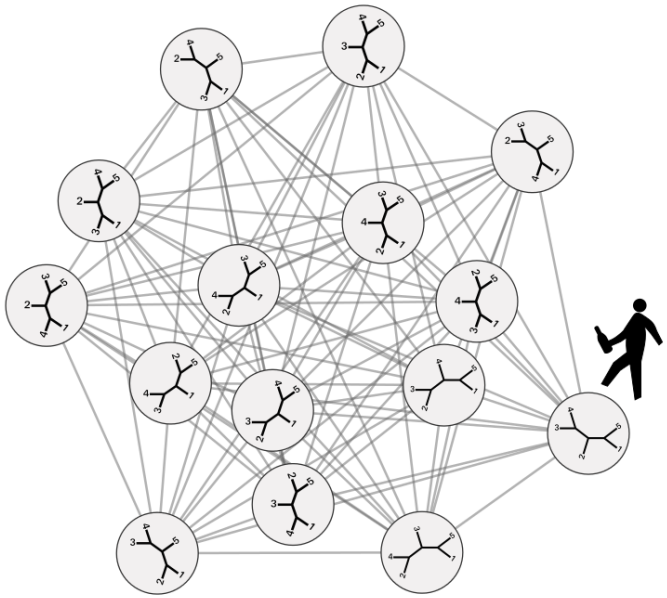
Likelihood

$$p(\boldsymbol{Y}|\tau, \boldsymbol{q}) = \prod_{i=1}^{M} \sum_{a^i} \eta(a^i_\rho) \prod_{(u,v)\in E(\tau)} P_{a^i_u a^i_v}(q_{uv})$$

Given a proper prior distribution $p(\tau, \boldsymbol{q})$, the posterior is

$$p(\tau, \boldsymbol{q}|\boldsymbol{Y}) \propto p(\boldsymbol{Y}|\tau, \boldsymbol{q}) p(\tau, \boldsymbol{q}).$$

Random-walk MCMC: use simple random perturbation (e.g., Nearest Neighborhood Interchange) to generate new state



Challenges

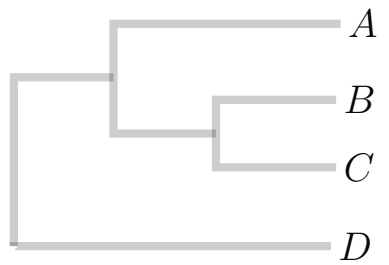
- Large search space (combinatorially exploding)

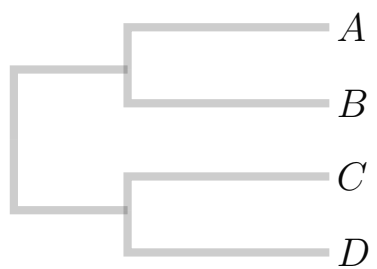
$$\#\text{ unrooted trees } (n \text{ taxa}) = (2n-5)!!$$

- Intertwined parameter space, low acceptance rate, hard to scale to data sets with many sequences.

Variational Bayesian Phylogenetic Inference

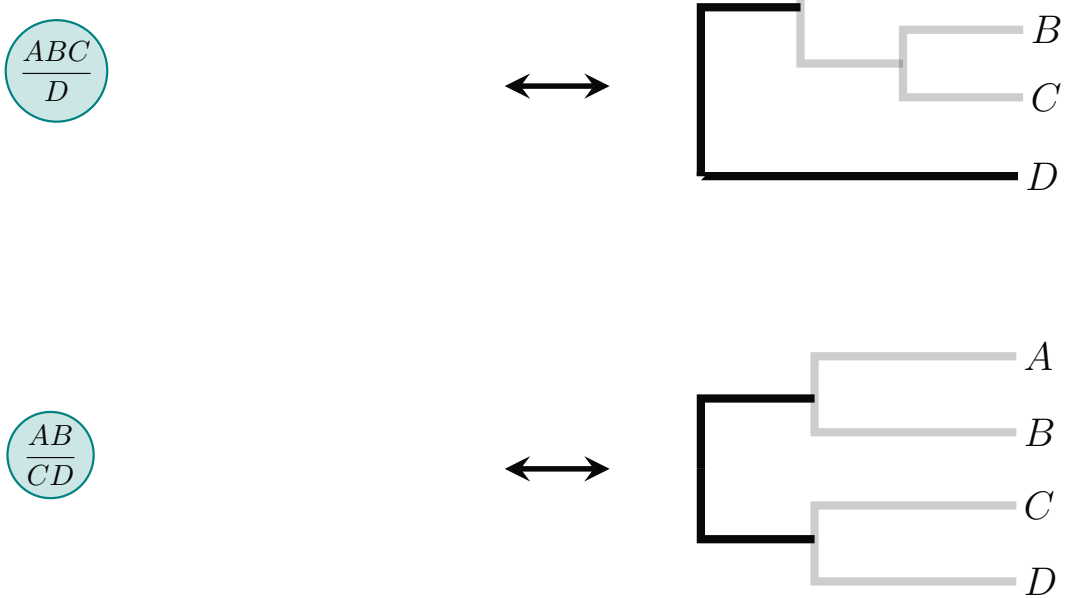
Encode tree structures via Bayesian networks!





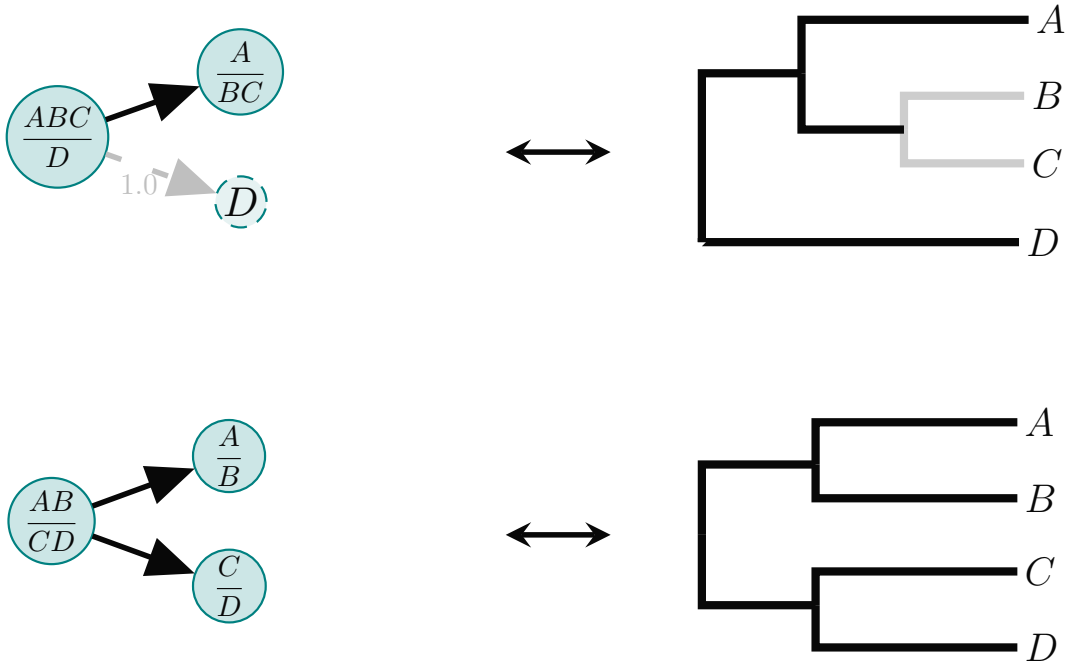
Subsplit Bayesian Networks

Encode tree structures via Bayesian networks!



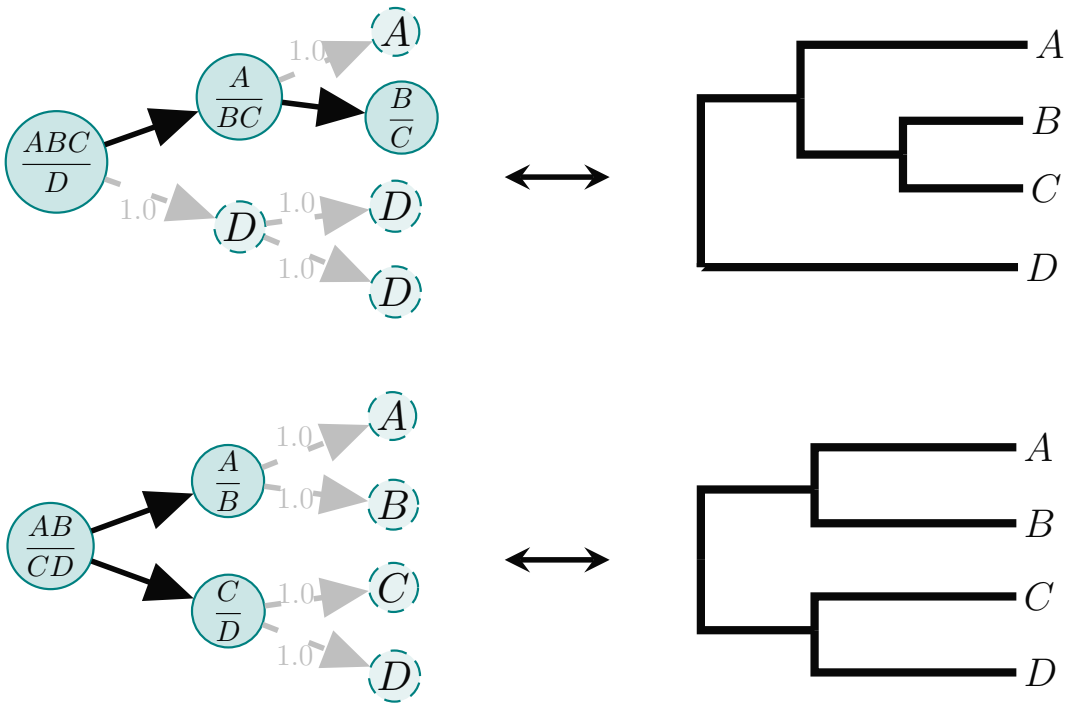
Subsplit Bayesian Networks

Encode tree structures via Bayesian networks!



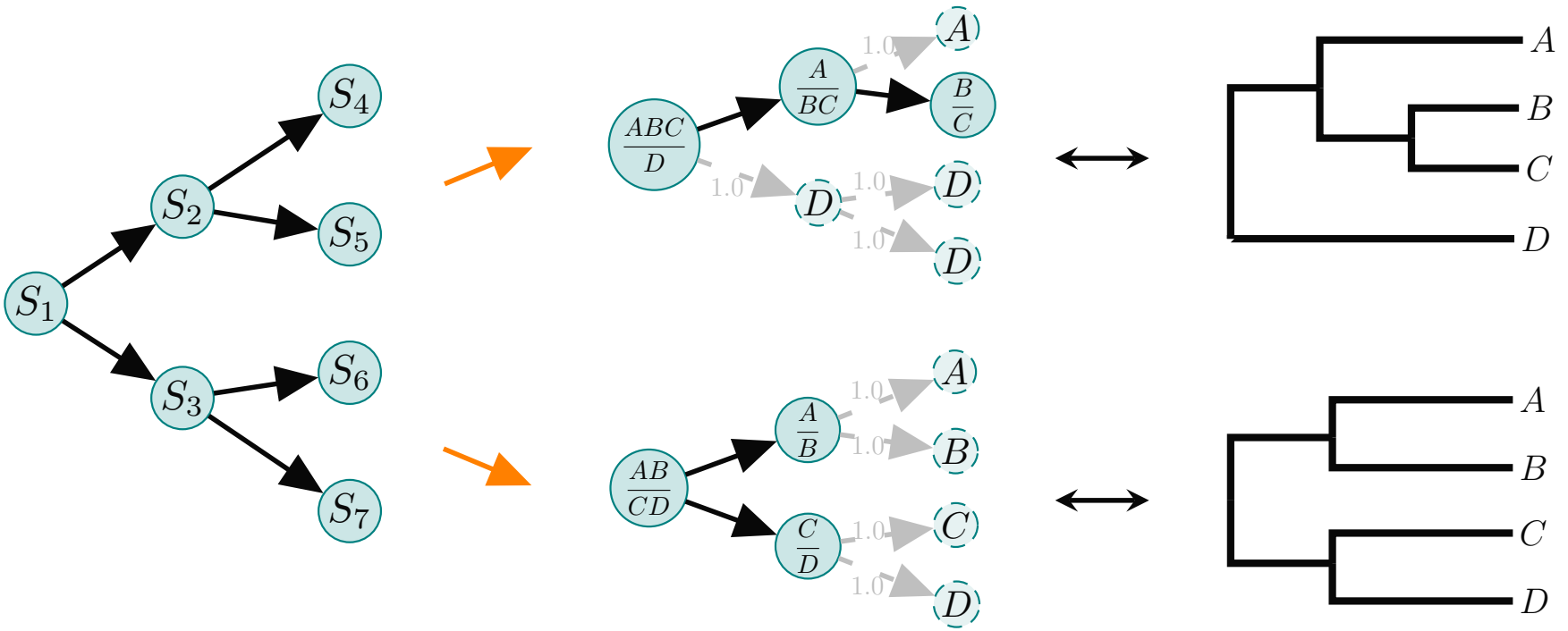
Subsplit Bayesian Networks

Encode tree structures via Bayesian networks!

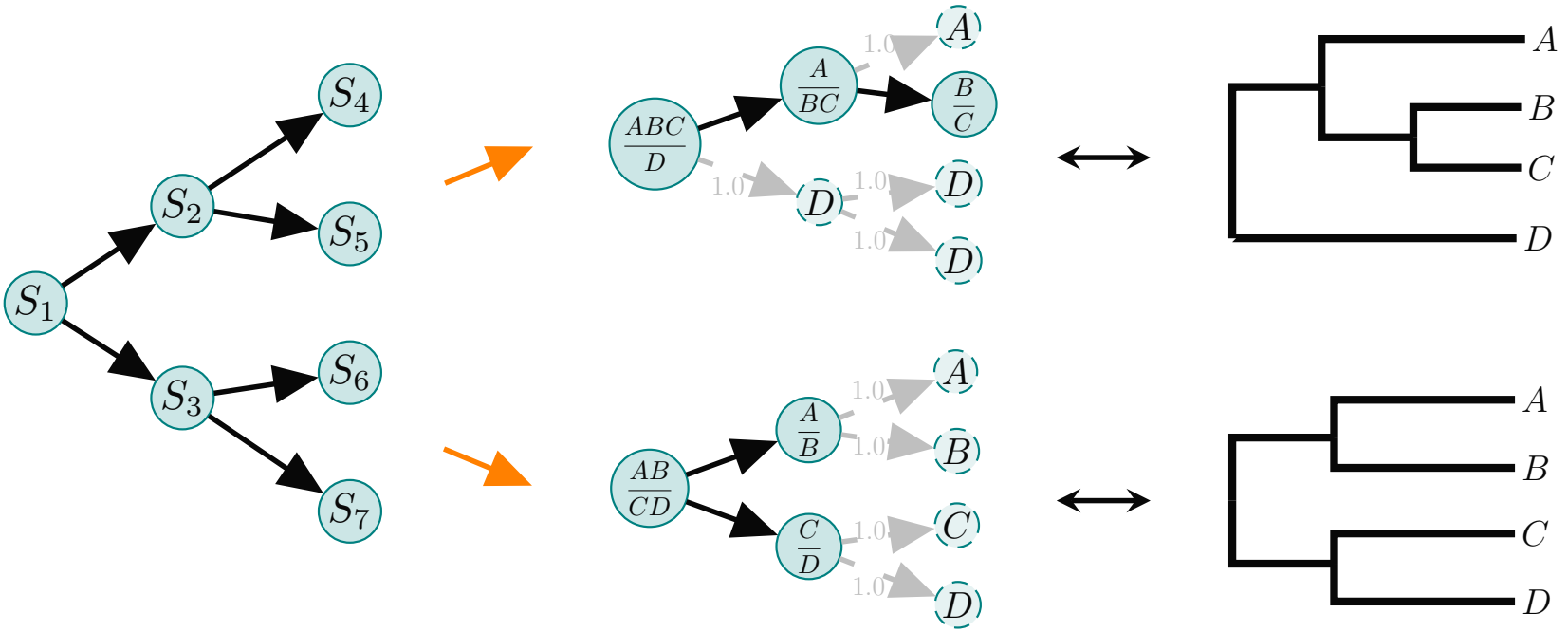


Subsplit Bayesian Networks

Encode tree structures via Bayesian networks!



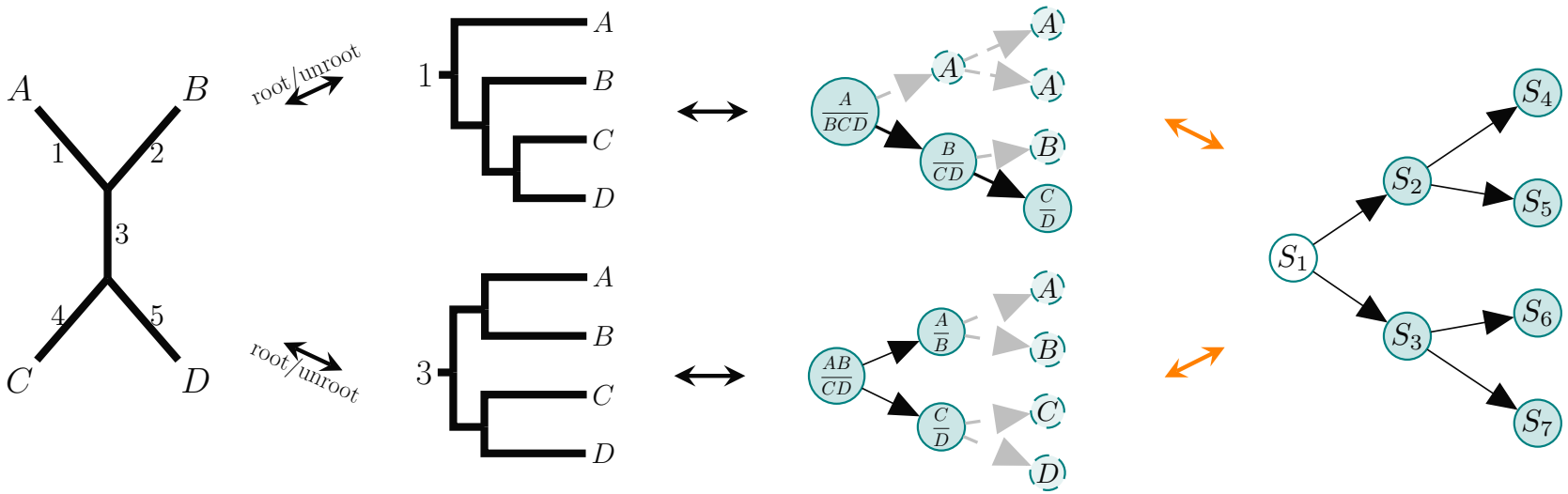
Probability Estimation Over Tree Topologies



Rooted Trees

$$p_{\text{sbn}}(T = \tau) = p(S_1 = s_1) \prod_{i>1} p(S_i = s_i | S_{\pi_i} = s_{\pi_i}).$$

Probability Estimation Over Tree Topologies



Unrooted Trees:

$$p_{\text{sbn}}(T^{\text{u}} = \tau) = \sum_{s_1 \sim \tau} p(S_1 = s_1) \prod_{i>1} p(S_i = s_i | S_{\pi_i} = s_{\pi_i}).$$

Tree Probability Estimation via SBNs

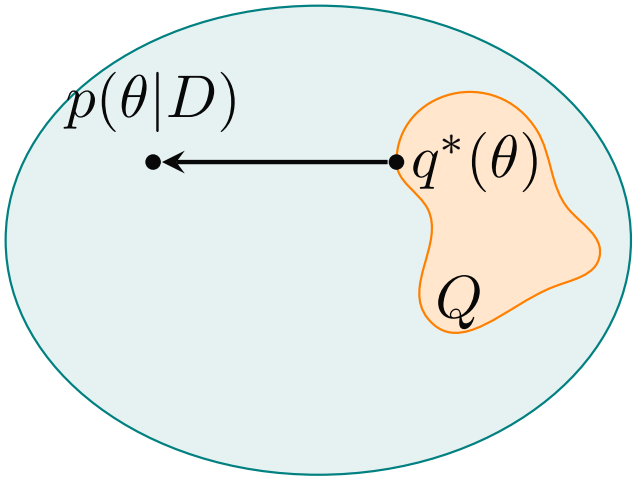
Use SBNs to refine posterior probability estimation given MCMC samples, useful when the posterior is diffuse.

Data set	(#Taxa, #Sites)	Tree space size	Sampled trees	KL divergence to ground truth				
				SRF	CCD	**SBN-SA**	**SBN-EM**	**SBN-EM-α**
DS1	(27, 1949)	5.84×10^{32}	1228	0.0155	0.6027	0.0687	0.0136	**0.0130**
DS2	(29, 2520)	1.58×10^{35}	7	**0.0122**	0.0218	0.0218	0.0199	0.0128
DS3	(36, 1812)	4.89×10^{47}	43	0.3539	0.2074	0.1152	0.1243	**0.0882**
DS4	(41, 1137)	1.01×10^{57}	828	0.5322	0.1952	0.1021	0.0763	**0.0637**
DS5	(50, 378)	2.84×10^{74}	33752	11.5746	1.3272	0.8952	0.8599	**0.8218**
DS6	(50, 1133)	2.84×10^{74}	35407	10.0159	0.4526	**0.2613**	0.3016	0.2786
DS7	(59, 1824)	4.36×10^{92}	1125	1.2765	0.3292	0.2341	0.0483	**0.0399**
DS8	(64, 1008)	1.04×10^{103}	3067	2.1653	0.4149	0.2212	0.1415	**0.1236**

[Zhang and Matsen, NeurIPS 2018]

Remark: Unlike previous methods, SBNs are flexible enough to provide accurate approximations to real data posteriors!

Variational Inference

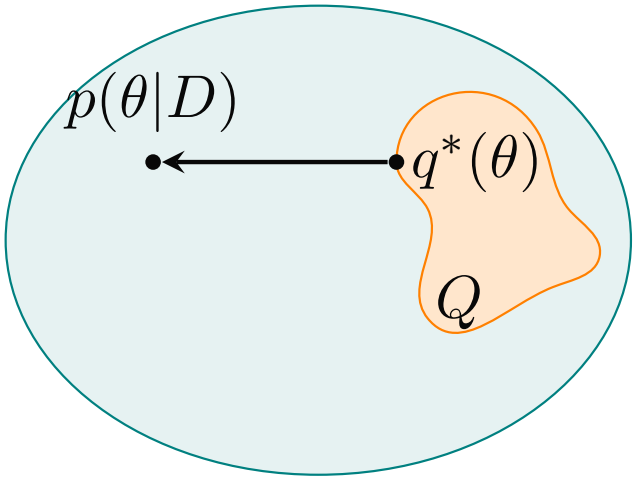


An optimization approach

$$q^* = \arg\min_{q \in Q} D_{KL}\left(q(\theta)||p(\theta|D)\right)$$

Q: approximating distributions.

Variational Inference



An optimization approach

$$q^* = \arg\min_{q \in Q} D_{KL}\left(q(\theta)||p(\theta|D)\right)$$

Q: approximating distributions.

equivalent to maximizing the evidence lower bound (ELBO)

$$L(q, D) = \mathbb{E}_{q(\theta)} \log\left(\frac{p(\theta, D)}{q(\theta)}\right) \leq \log p(D)$$

- tends to be faster than MCMC
- easy to scale to large data sets (via stochastic gradient ascent)

- Approximating Distribution:

tree topology

$$Q_\phi(\tau)$$

- Approximating Distribution:

$$Q_{\phi,\psi}(\tau, q) \triangleq \underbrace{Q_\phi(\tau)}_{\text{tree topology}} \cdot \underbrace{Q_\psi(q|\tau)}_{\text{branch length}}$$

- Approximating Distribution:

$$Q_{\phi,\psi}(\tau, q) \triangleq \underbrace{Q_\phi(\tau)}_{\text{tree topology}} \cdot \underbrace{Q_\psi(q|\tau)}_{\text{branch length}}$$

- Multi-sample Lower Bound:

$$L^K(\boldsymbol{\phi}, \boldsymbol{\psi}) = \mathbb{E}_{Q_{\boldsymbol{\phi},\boldsymbol{\psi}}(\tau^{1:K}, \boldsymbol{q}^{1:K})} \log \left(\frac{1}{K} \sum_{i=1}^{K} \frac{p(\boldsymbol{Y}|\tau^i, \boldsymbol{q}^i) p(\tau^i, \boldsymbol{q}^i)}{Q_{\boldsymbol{\phi}}(\tau^i) Q_{\boldsymbol{\psi}}(\boldsymbol{q}^i|\tau^i)} \right)$$

- Approximating Distribution:

$$Q_{\phi,\psi}(\tau, q) \triangleq \overbrace{Q_\phi(\tau)}^{\text{tree topology}} \cdot \overbrace{Q_\psi(q|\tau)}^{\text{branch length}}$$

- Multi-sample Lower Bound:

$$L^K(\boldsymbol{\phi}, \boldsymbol{\psi}) = \mathbb{E}_{Q_{\boldsymbol{\phi},\boldsymbol{\psi}}(\tau^{1:K}, \boldsymbol{q}^{1:K})} \log \left(\frac{1}{K} \sum_{i=1}^{K} \frac{p(\boldsymbol{Y}|\tau^i, \boldsymbol{q}^i) p(\tau^i, \boldsymbol{q}^i)}{Q_{\boldsymbol{\phi}}(\tau^i) Q_{\boldsymbol{\psi}}(\boldsymbol{q}^i|\tau^i)} \right)$$

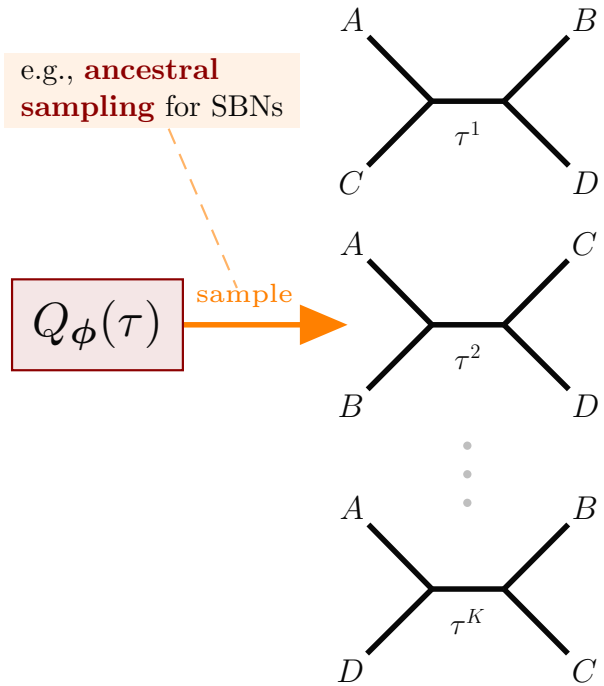
- **Variational Bayesian Phylogenetic Inference**:

$$\hat{\boldsymbol{\phi}}, \hat{\boldsymbol{\psi}} = \arg\max_{\boldsymbol{\phi},\boldsymbol{\psi}} L^K(\boldsymbol{\phi}, \boldsymbol{\psi})$$

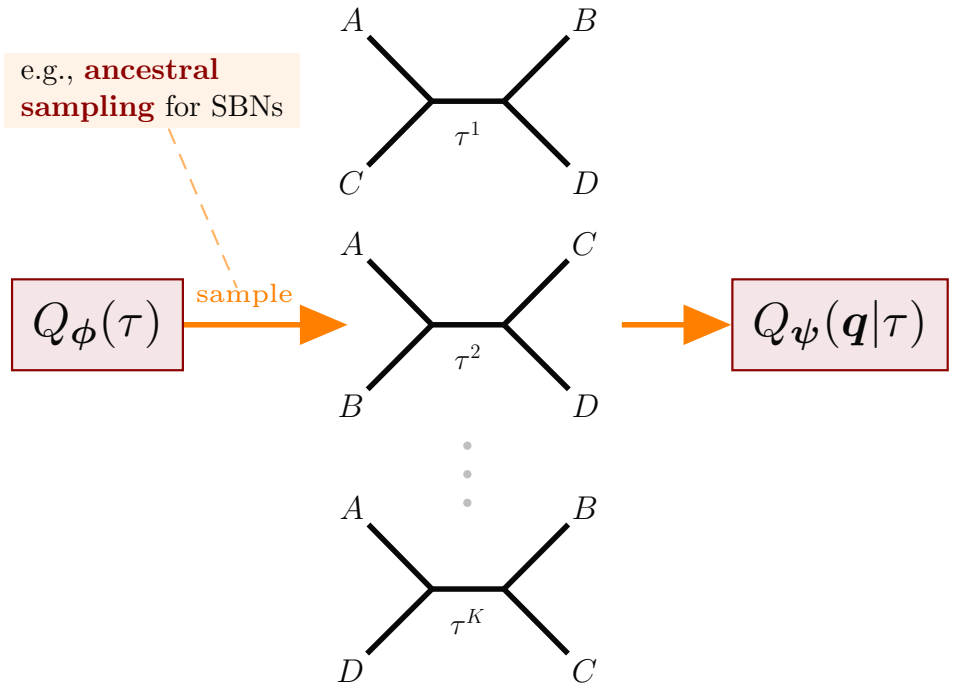
parameters trained via stochastic gradient ascent (SGA).

$Q_\phi(\tau)$

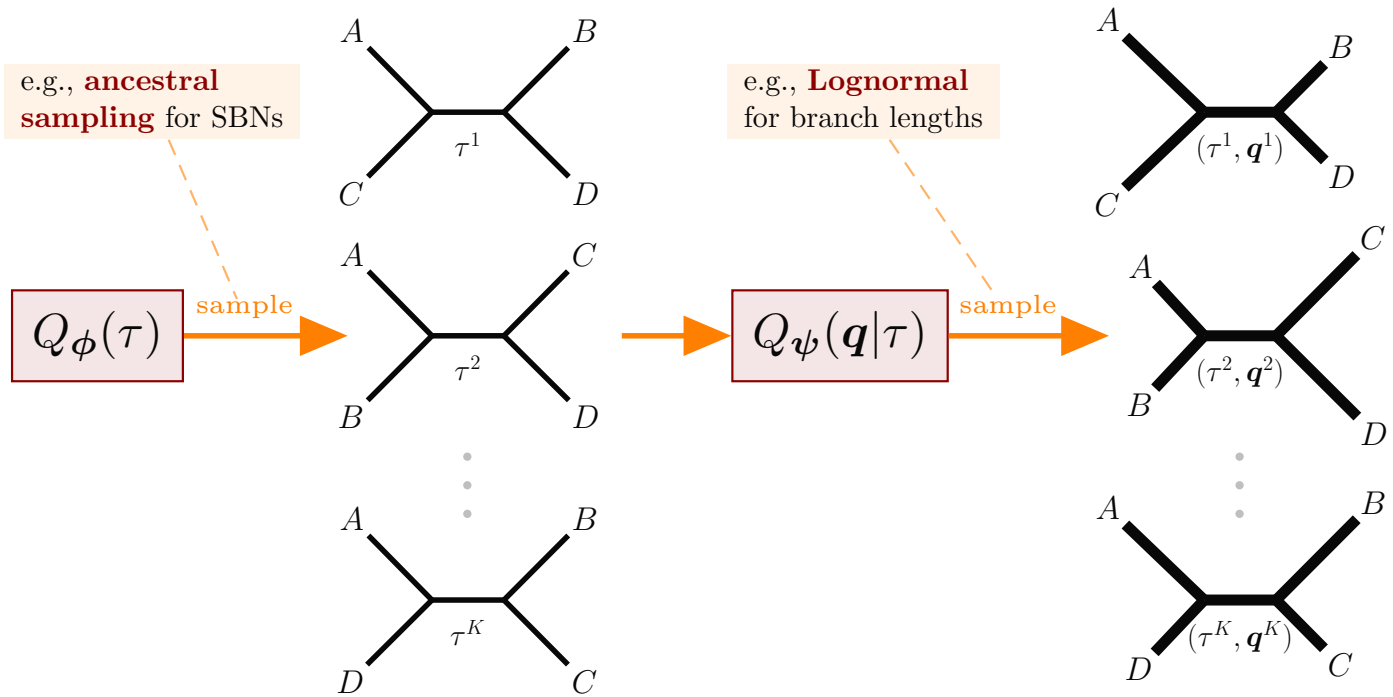
Variational Bayesian Phylogenetic Inference



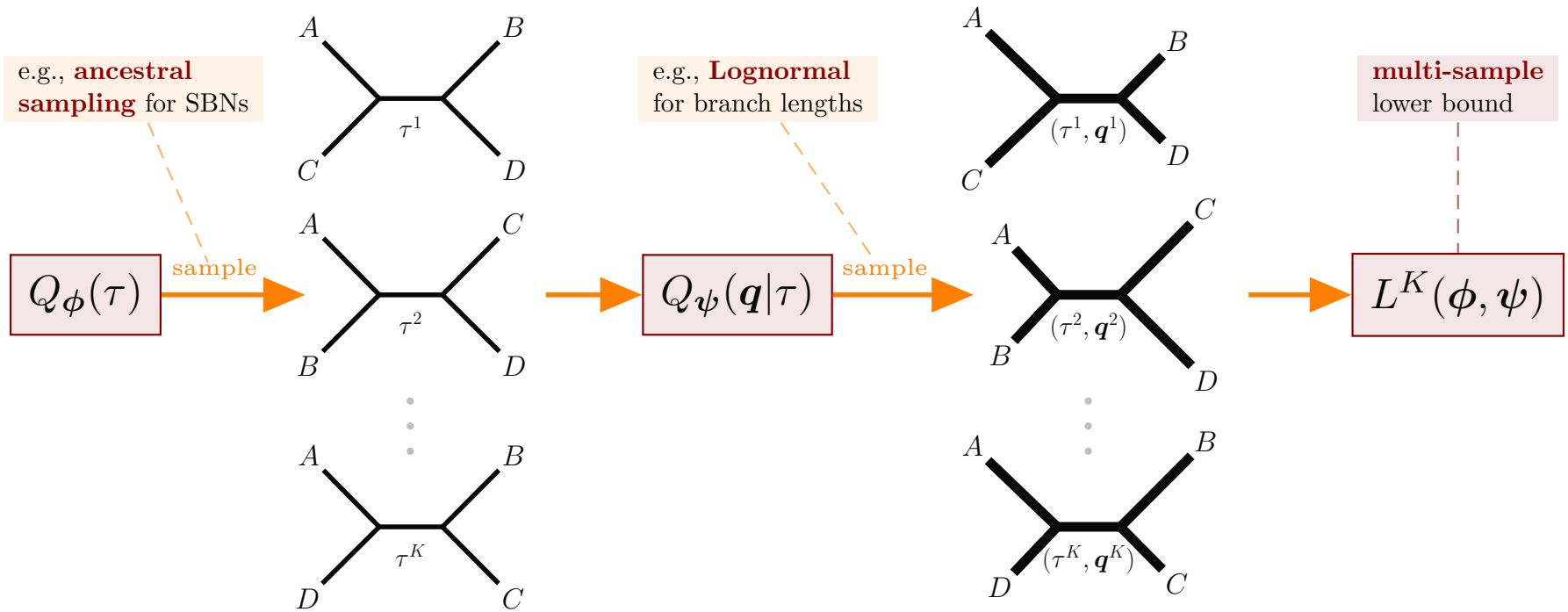
Variational Bayesian Phylogenetic Inference



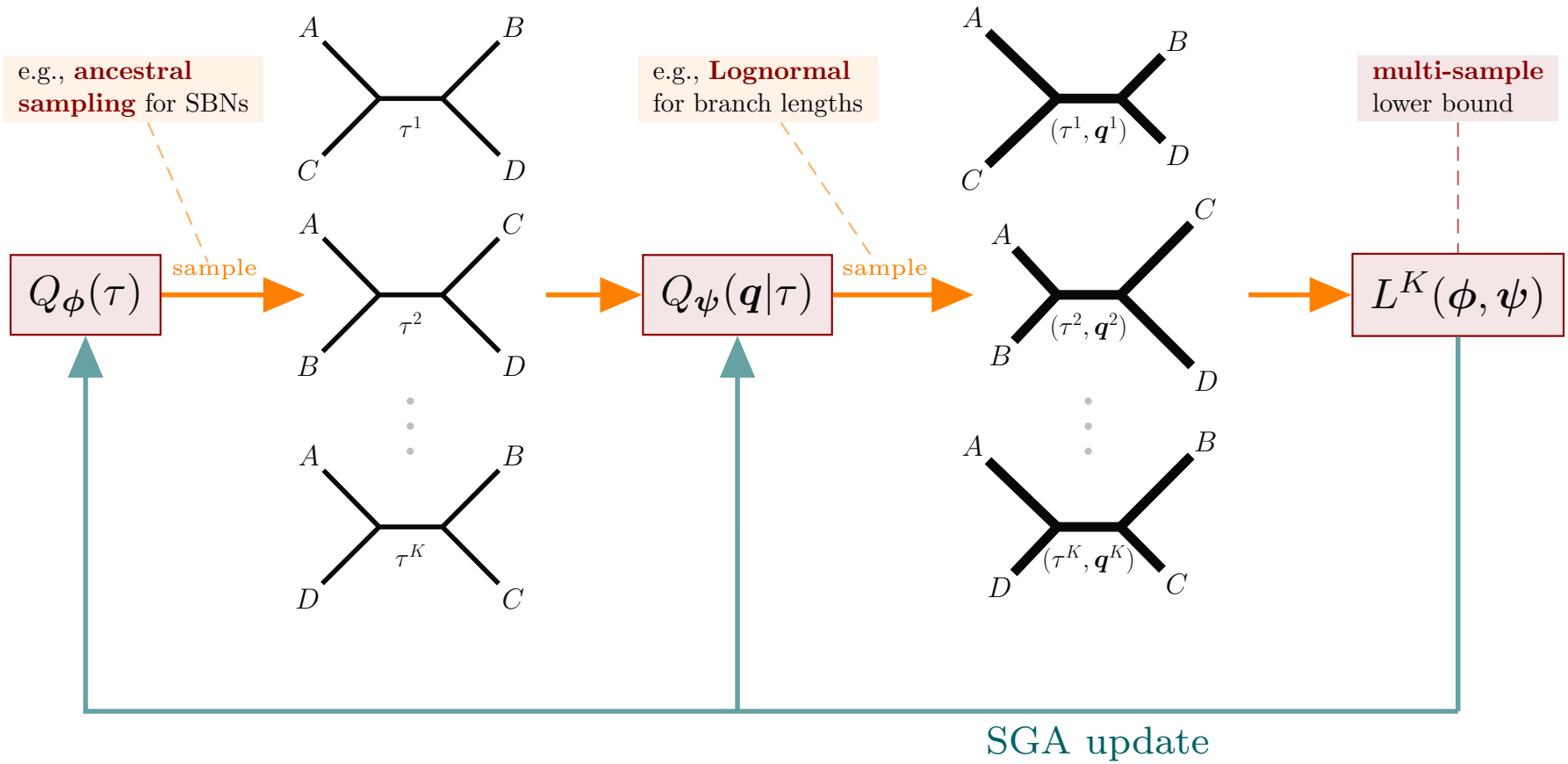
Variational Bayesian Phylogenetic Inference



Variational Bayesian Phylogenetic Inference



Variational Bayesian Phylogenetic Inference



Structured Parameterization

SBNs Parameters

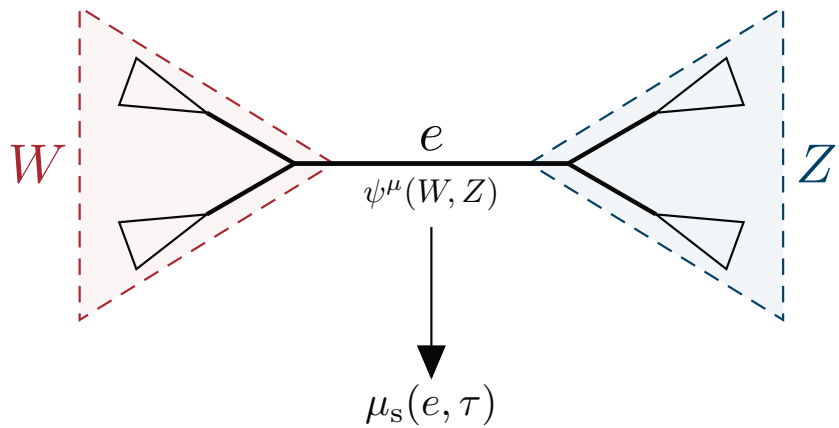
$$p(S_1 = s_1) = \frac{\exp(\phi_{s_1})}{\sum_{s_r \in \mathbb{S}_r} \exp(\phi_{s_r})}, \quad p(S_i = s | S_{\pi_i} = t) = \frac{\exp(\phi_{s|t})}{\sum_{s \in \mathbb{S}_{\cdot|t}} \exp(\phi_{s|t})}$$

Branch Length Parameters

$$Q_{\boldsymbol{\psi}}(\boldsymbol{q}|\tau) = \prod_{e \in E(\tau)} p^{\text{Lognormal}}\left(q_e \mid \mu(e,\tau), \sigma(e,\tau)\right)$$

- *Simple Split*

$$\mu_{\text{s}}(e,\tau) = \psi^{\mu}_{e/\tau}, \ \sigma_{\text{s}}(e,\tau) = \psi^{\sigma}_{e/\tau}.$$



Structured Parameterization

SBNs Parameters

$$p(S_1 = s_1) = \frac{\exp(\phi_{s_1})}{\sum_{s_r \in \mathbb{S}_r} \exp(\phi_{s_r})}, \quad p(S_i = s | S_{\pi_i} = t) = \frac{\exp(\phi_{s|t})}{\sum_{s \in \mathbb{S}_{\cdot|t}} \exp(\phi_{s|t})}$$

Branch Length Parameters

$$Q_{\boldsymbol{\psi}}(\boldsymbol{q}|\tau) = \prod_{e \in E(\tau)} p^{\text{Lognormal}}\left(q_e \mid \mu(e,\tau), \sigma(e,\tau)\right)$$

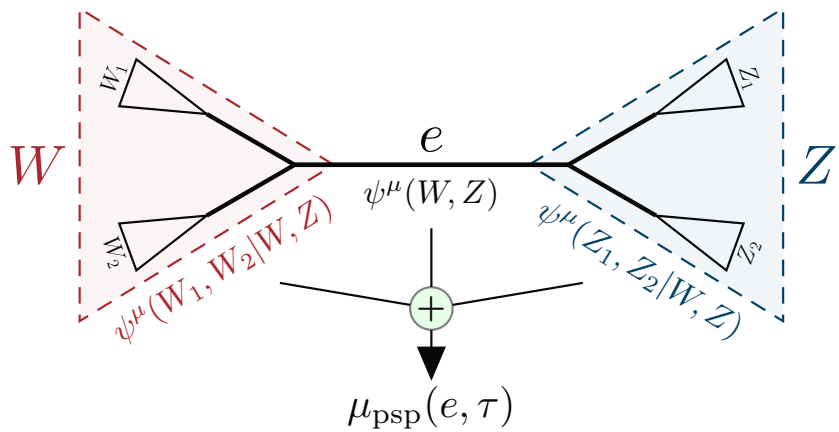
- *Simple Split*

  $$\mu_{\text{s}}(e,\tau) = \psi^{\mu}_{e/\tau}, \ \sigma_{\text{s}}(e,\tau) = \psi^{\sigma}_{e/\tau}.$$

- *Primary Subsplit Pair* (PSP)

  $$\mu_{\text{psp}}(e,\tau) = \psi^{\mu}_{e/\tau} + \sum_{s \in e /\!\!/ \tau} \psi^{\mu}_{s}$$

  $$\sigma_{\text{psp}}(e,\tau) = \psi^{\sigma}_{e/\tau} + \sum_{s \in e /\!\!/ \tau} \psi^{\sigma}_{s}.$$



Stochastic Gradient Estimators

SBNs Parameters $\boldsymbol{\phi}$. With $\tau^j, \boldsymbol{q}^j \overset{\text{iid}}{\sim} Q_{\boldsymbol{\phi},\boldsymbol{\psi}}(\tau,\ \boldsymbol{q})$

- *VIMCO*. [Minh and Rezende, ICML 2016]

$$\nabla_{\boldsymbol{\phi}} L^K(\boldsymbol{\phi},\boldsymbol{\psi}) \simeq \sum_{j=1}^{K} \left(\hat{L}^K_{j|-j}(\boldsymbol{\phi},\boldsymbol{\psi}) - \tilde{w}^j\right) \nabla_{\boldsymbol{\phi}} \log Q_{\boldsymbol{\phi}}(\tau^j).$$

- *RWS*. [Bornschein and Bengio, ICLR 2015]

$$\nabla_{\boldsymbol{\phi}} L^K(\boldsymbol{\phi},\boldsymbol{\psi}) \simeq \sum_{j=1}^{K} \tilde{w}^j \nabla_{\boldsymbol{\phi}} \log Q_{\boldsymbol{\phi}}(\tau^j).$$

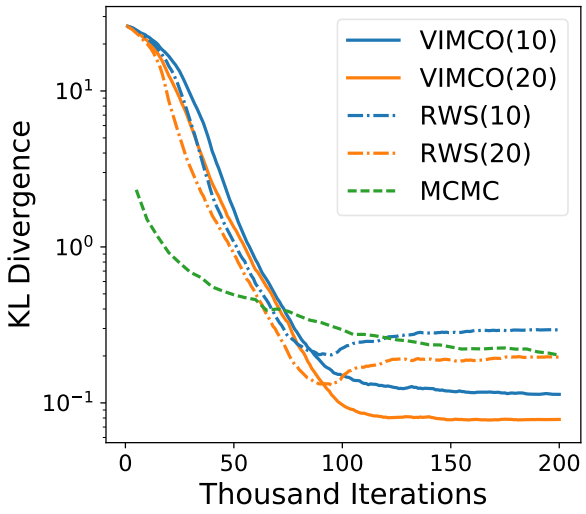
Branch Length Parameters $\boldsymbol{\psi}$. $g_{\boldsymbol{\psi}}(\boldsymbol{\epsilon}|\tau) = \exp(\boldsymbol{\mu}_{\boldsymbol{\psi},\tau} + \boldsymbol{\sigma}_{\boldsymbol{\psi},\tau} \odot \boldsymbol{\epsilon})$.

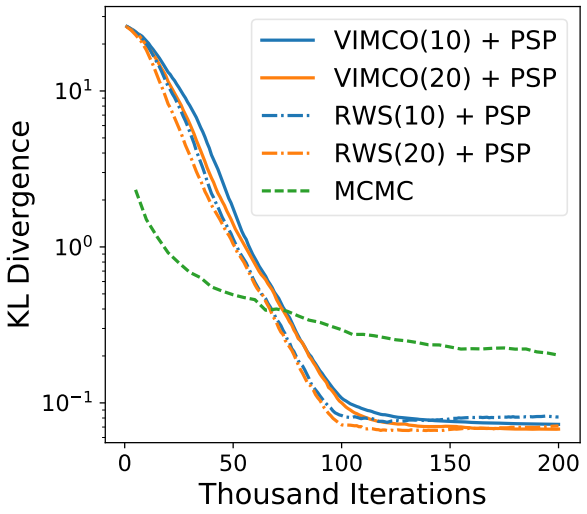
- *Reparameterization Trick*. Let $f_{\boldsymbol{\phi},\boldsymbol{\psi}}(\tau,\boldsymbol{q}) = \frac{p(\boldsymbol{Y}|\tau,\boldsymbol{q})p(\tau,\boldsymbol{q})}{Q_{\boldsymbol{\phi}}(\tau)Q_{\boldsymbol{\psi}}(\boldsymbol{q}|\tau)}$.

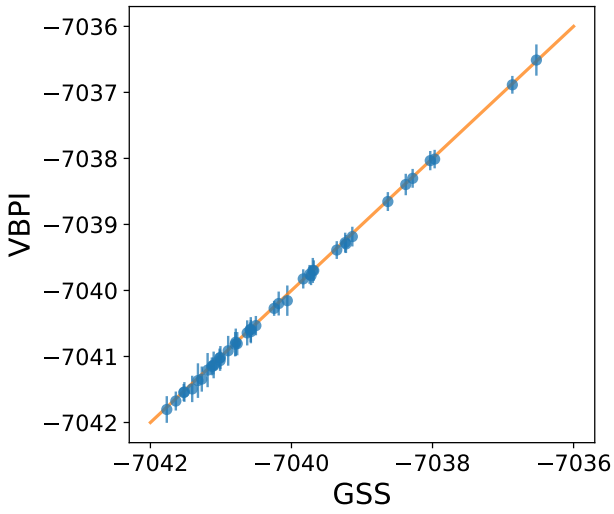
$$\nabla_{\boldsymbol{\psi}} L^K(\boldsymbol{\phi},\boldsymbol{\psi}) \simeq \sum_{j=1}^{K} \tilde{w}^j \nabla_{\boldsymbol{\psi}} \log f_{\boldsymbol{\phi},\boldsymbol{\psi}}(\tau^j, g_{\boldsymbol{\psi}}(\boldsymbol{\epsilon}^j|\tau^j))$$

where $\tau^j \overset{\text{iid}}{\sim} Q_{\boldsymbol{\phi}}(\tau)$, $\boldsymbol{\epsilon}^j \overset{\text{iid}}{\sim} \mathcal{N}(\boldsymbol{0},\boldsymbol{I})$.

Performance on Real Data

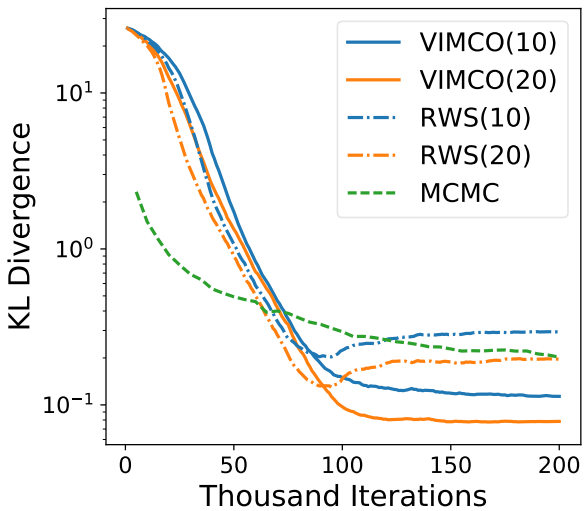


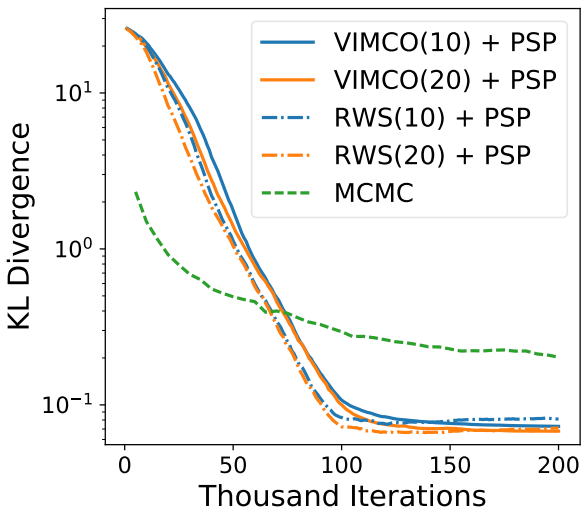


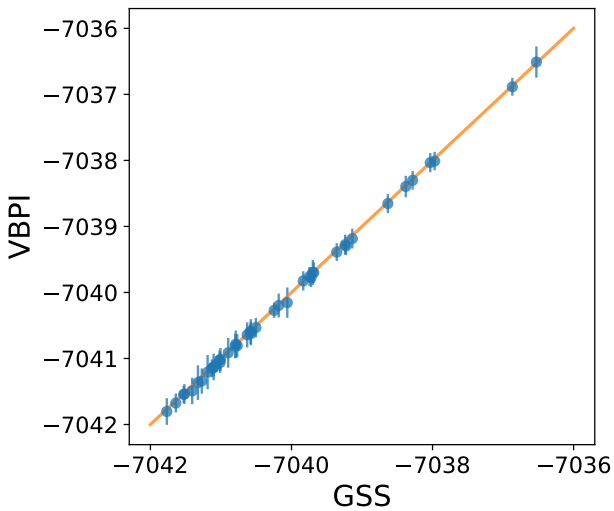


[Zhang and Matsen, ICLR 2019]

Performance on Real Data



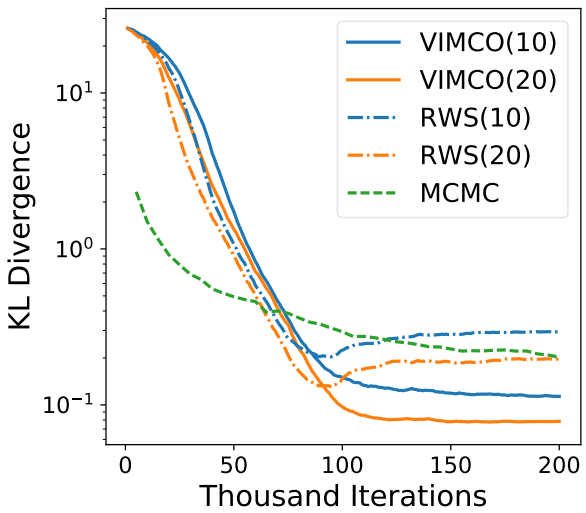


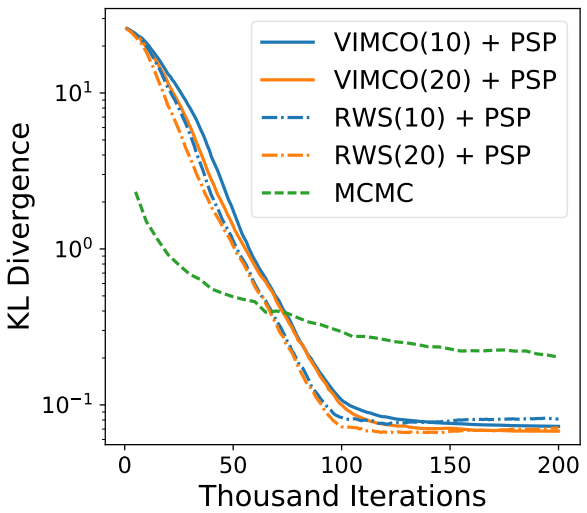


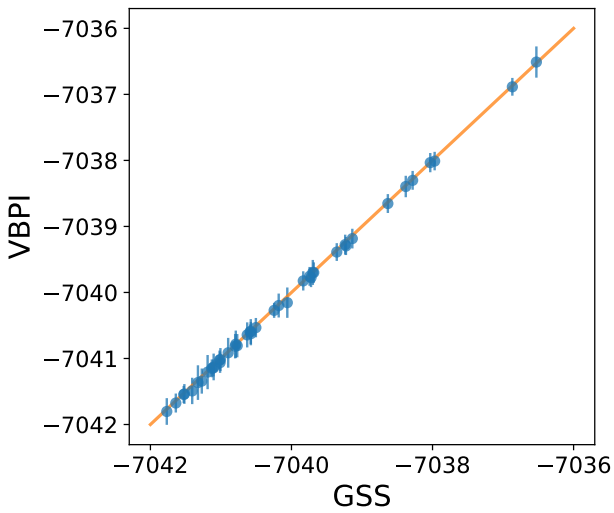
[Zhang and Matsen, ICLR 2019]

More samples $\Rightarrow$ better exploration $\Rightarrow$ better approximation

Performance on Real Data





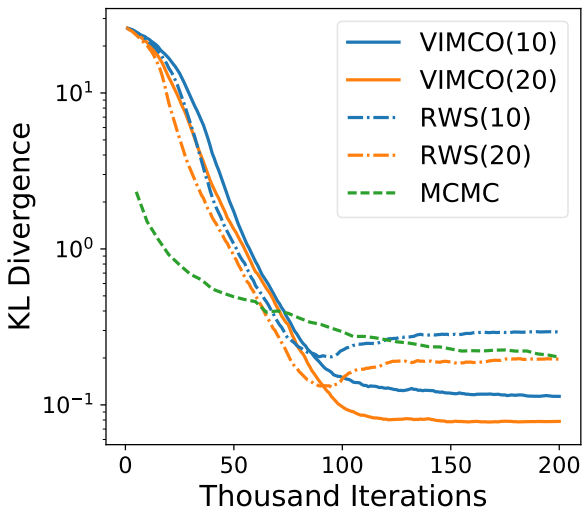


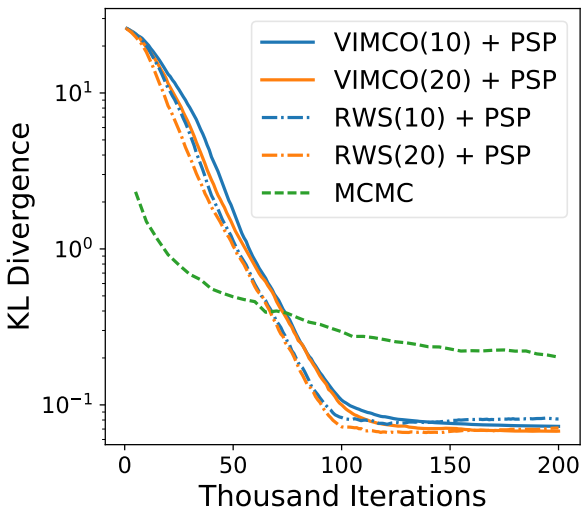
[Zhang and Matsen, ICLR 2019]

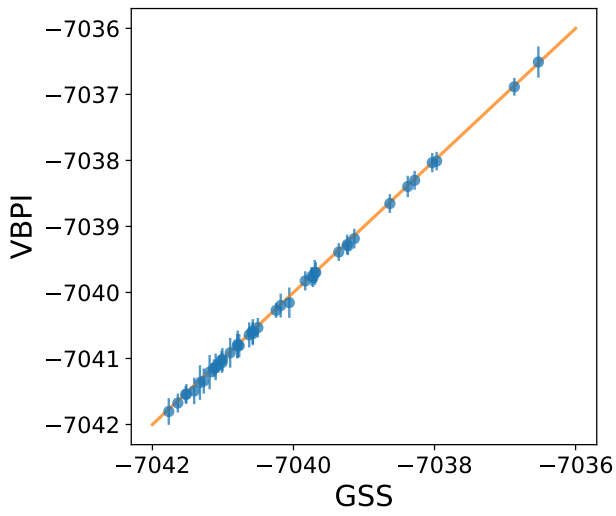
More samples ⇒ better exploration ⇒ better approximation

More flexible branch length distributions across tree topologies (PSP) ease training and improve approximation

Performance on Real Data







[Zhang and Matsen, ICLR 2019]

More samples ⇒ better exploration ⇒ better approximation

More flexible branch length distributions across tree topologies (PSP) ease training and improve approximation

Outperform MCMC via much more efficient tree space exploration and branch length updates

Performance on Real Data

Data set	Marginal Likelihood (NATs)				
	VIMCO(10)	VIMCO(20)	VIMCO(10)+PSP	VIMCO(20)+PSP	SS
DS1	-7108.43(0.26)	-7108.35(0.21)	-7108.41(0.16)	**-7108.42(0.10)**	-7108.42(0.18)
DS2	-26367.70(0.12)	-26367.71(0.09)	**-26367.72(0.08)**	-26367.70(0.10)	-26367.57(0.48)
DS3	-33735.08(0.11)	-33735.11(0.11)	**-33735.10(0.09)**	-33735.07(0.11)	-33735.44(0.50)
DS4	-13329.90(0.31)	-13329.98(0.20)	**-13329.94(0.18)**	-13329.93(0.22)	-13330.06(0.54)
DS5	-8214.36(0.67)	-8214.74(0.38)	-8214.61(0.38)	-8214.55(0.43)	**-8214.51(0.28)**
DS6	-6723.75(0.68)	-6723.71(0.65)	-6724.09(0.55)	**-6724.34(0.45)**	-6724.07(0.86)
DS7	-37332.03(0.43)	-37331.90(0.49)	-37331.90(0.32)	**-37332.03(0.23)**	-37332.76(2.42)
DS8	-8653.34(0.55)	-8651.54(0.80)	**-8650.63(0.42)**	-8650.55(0.46)	-8649.88(1.75)

[Zhang and Matsen, ICLR 2019]

Performance on Real Data

Data set	Marginal Likelihood (NATs)				
	VIMCO(10)	VIMCO(20)	VIMCO(10)+PSP	VIMCO(20)+PSP	SS
DS1	-7108.43(0.26)	-7108.35(0.21)	-7108.41(0.16)	**-7108.42(0.10)**	-7108.42(0.18)
DS2	-26367.70(0.12)	-26367.71(0.09)	**-26367.72(0.08)**	-26367.70(0.10)	-26367.57(0.48)
DS3	-33735.08(0.11)	-33735.11(0.11)	**-33735.10(0.09)**	-33735.07(0.11)	-33735.44(0.50)
DS4	-13329.90(0.31)	-13329.98(0.20)	**-13329.94(0.18)**	-13329.93(0.22)	-13330.06(0.54)
DS5	-8214.36(0.67)	-8214.74(0.38)	-8214.61(0.38)	-8214.55(0.43)	**-8214.51(0.28)**
DS6	-6723.75(0.68)	-6723.71(0.65)	-6724.09(0.55)	**-6724.34(0.45)**	-6724.07(0.86)
DS7	-37332.03(0.43)	-37331.90(0.49)	-37331.90(0.32)	**-37332.03(0.23)**	-37332.76(2.42)
DS8	-8653.34(0.55)	-8651.54(0.80)	**-8650.63(0.42)**	-8650.55(0.46)	-8649.88(1.75)

[Zhang and Matsen, ICLR 2019]

Competitive to state-of-the-art (stepping-stone), dramatically reducing cost at test time: VBPI(1000) vs SS(100,000)

Performance on Real Data

Data set	Marginal Likelihood (NATs)				
	VIMCO(10)	VIMCO(20)	VIMCO(10)+PSP	VIMCO(20)+PSP	SS
DS1	-7108.43(0.26)	-7108.35(0.21)	-7108.41(0.16)	**-7108.42(0.10)**	-7108.42(0.18)
DS2	-26367.70(0.12)	-26367.71(0.09)	**-26367.72(0.08)**	-26367.70(0.10)	-26367.57(0.48)
DS3	-33735.08(0.11)	-33735.11(0.11)	**-33735.10(0.09)**	-33735.07(0.11)	-33735.44(0.50)
DS4	-13329.90(0.31)	-13329.98(0.20)	**-13329.94(0.18)**	-13329.93(0.22)	-13330.06(0.54)
DS5	-8214.36(0.67)	-8214.74(0.38)	-8214.61(0.38)	-8214.55(0.43)	**-8214.51(0.28)**
DS6	-6723.75(0.68)	-6723.71(0.65)	-6724.09(0.55)	**-6724.34(0.45)**	-6724.07(0.86)
DS7	-37332.03(0.43)	-37331.90(0.49)	-37331.90(0.32)	**-37332.03(0.23)**	-37332.76(2.42)
DS8	-8653.34(0.55)	-8651.54(0.80)	**-8650.63(0.42)**	-8650.55(0.46)	-8649.88(1.75)

[Zhang and Matsen, ICLR 2019]

Competitive to state-of-the-art (stepping-stone), dramatically reducing cost at test time: VBPI(1000) vs SS(100,000)

PSP alleviates the demand for large samples, reducing computation while maintaining approximation accuracy

Conclusion

- We introduced **VBPI**, a general variational framework for Bayesian phylogenetic inference.
- **VBPI** allows efficient learning on both tree topology and branch lengths, providing competitive performance to MCMC while requiring much less computation.
- Can be used for further statistical analysis (e.g., marginal likelihood estimation) via importance sampling.

[1] **Zhang, C.** and Matsen F. A., Generalizing Tree Probability Estimation via Bayesian Networks. In *Advances in Neural Information Processing Systems*, **spotlight**(3.5%), 2018.

[2] **Zhang, C.** and Matsen F. A., Variational Bayesian Phylogenetic Inference. In *Proceedings of the 7th International Conference on Learning Representations*, 2019.

Thank you!