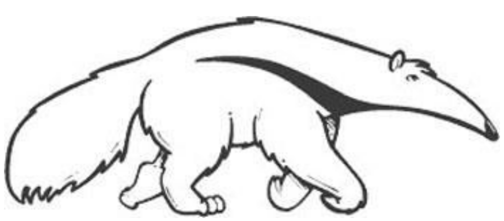
Anytime Approximate Inference in Graphical Models

Qi Lou

Final Defense

Dec. 5, 2018

Committee:
Alexander Ihler (Chair)
Rina Dechter
Sameer Singh



BREN:ICS INFORMATION AND COMPUTER SCIENCES

UNIVERSITY of CALIFORNIA IRVINE

Core of This Thesis

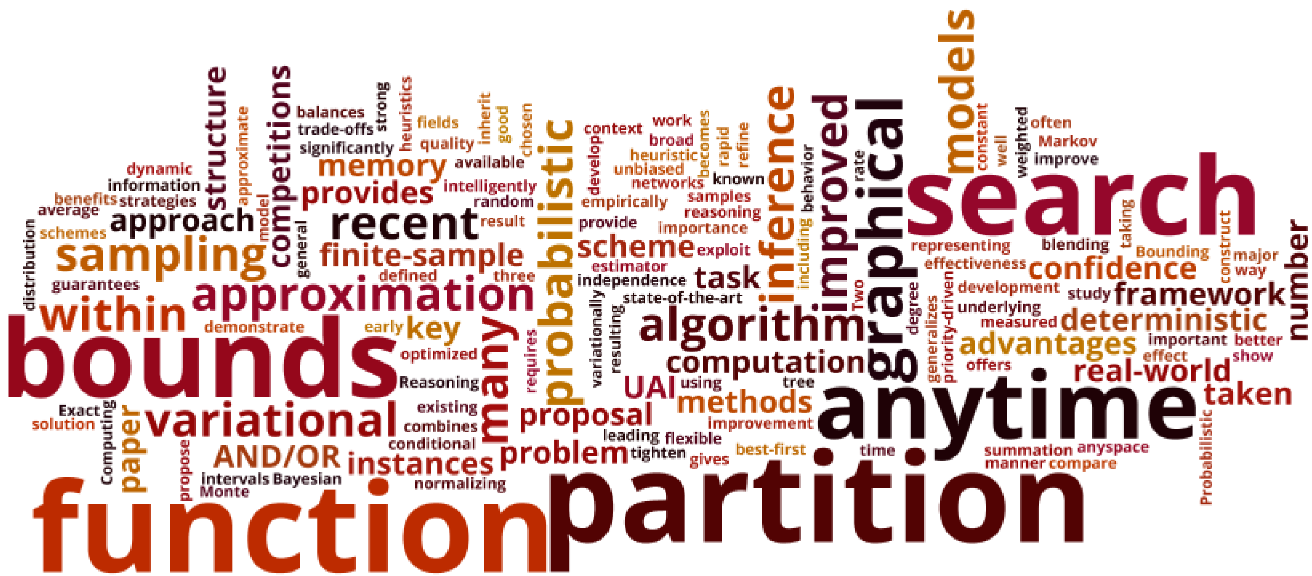
Qi Lou, Rina Dechter, Alexander Ihler. Interleave variational optimization with Monte Carlo sampling: A tale of two approximate inference paradigms. *AAAI Conference on Artificial Intelligence (AAAI)*, 2019. To appear.

Qi Lou, Rina Dechter, Alexander Ihler. Finite-sample bounds for marginal MAP. *Uncertainty in Artificial Intelligence (UAI)*, 2018.

Qi Lou, Rina Dechter, Alexander Ihler. Anytime anyspace AND/OR search for bounding marginal MAP. *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

Qi Lou, Rina Dechter, Alexander Ihler. Dynamic importance sampling for anytime bounds of the partition function. *Neural Information Processing Systems (NIPS)*, 2017.

Qi Lou, Rina Dechter, Alexander Ihler. Anytime anyspace AND/OR search for bounding the partition function. *AAAI Conference on Artificial Intelligence (AAAI)*, 2017.



Graphical Models

- Describe structure in large problems
  - Large complex system $f(X)$
  - Made of "smaller", "local" interactions $f_\alpha(X_\alpha)$
  - Complexity emerges through interdependence
- More formally:

A *graphical model* consists of:

$X = \{X_1, \ldots, X_n\}$ -- variables (we'll assume discrete)

$D = \{D_1, \ldots, D_n\}$ -- domains

$F = \{f_{\alpha_1}, \ldots, f_{\alpha_m}\}$ -- (non-negative) functions or "factors"

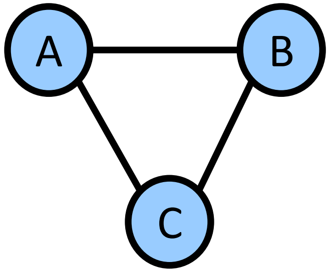
- Example:

$$f(A,B,C) = f(A,B)f(A,C)f(B,C)$$

A	B	f(A,B)
0	0	**0.24**
0	1	**0.56**
1	0	**1.1**
1	1	**1.2**

…

B	C	f(B,C)
0	0	**0.12**
0	1	**0.36**
1	0	**0.3**
1	1	**1.8**

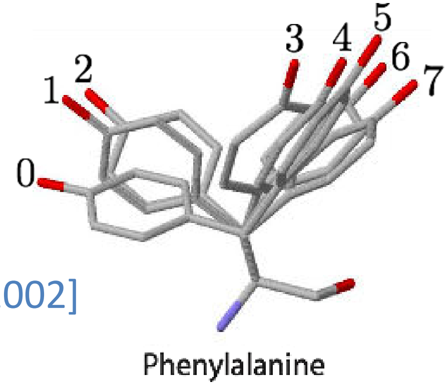


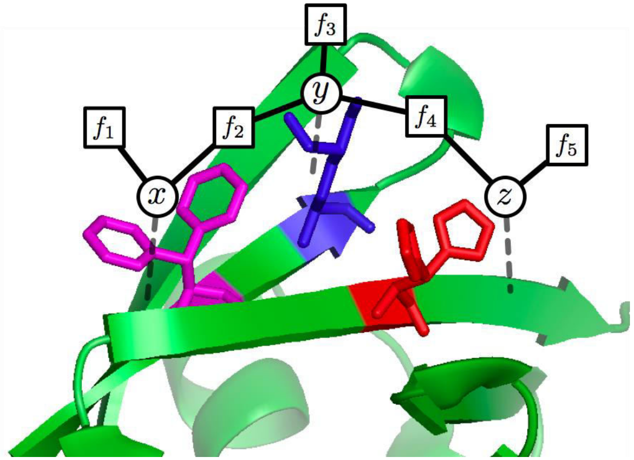
Graphical Models

- Describe structure in large problems
  - Large complex system $f(X)$
  - Made of "smaller", "local" interactions $f_\alpha(X_\alpha)$
  - Complexity emerges through interdependence

- Examples & Tasks
  - Maximization (**MAP**): compute the most probable configuration

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} \prod_\alpha f_\alpha(\mathbf{x}_\alpha) \qquad f(\mathbf{x}^*) = \max_{\mathbf{x}} \prod_\alpha f_\alpha(\mathbf{x}_\alpha)$$





[Yanover & Weiss 2002]

Graphical Models

- Describe structure in large problems
  - Large complex system $f(X)$
  - Made of "smaller", "local" interactions $f_\alpha(X_\alpha)$
  - Complexity emerges through interdependence

- Examples & Tasks
  - Summation & marginalization

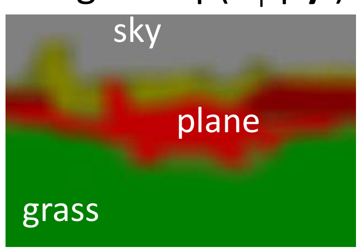
"partition function"

$$p(x_i) = \frac{1}{Z} \sum_{\mathbf{x} \setminus x_i} \prod_\alpha f_\alpha(\mathbf{x}_\alpha) \quad \text{and} \quad Z = \sum_{\mathbf{x}} \prod_\alpha f_\alpha(\mathbf{x}_\alpha)$$

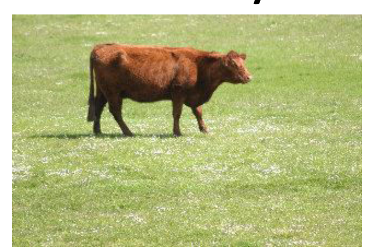
Observation **y**

$\Rightarrow$

Marginals p(x_i | **y**)

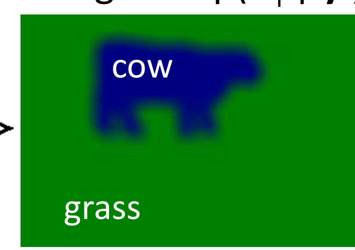


Observation **y**



$\Rightarrow$

Marginals p(x_i | **y**)



e.g., [Plath et al. 2009]

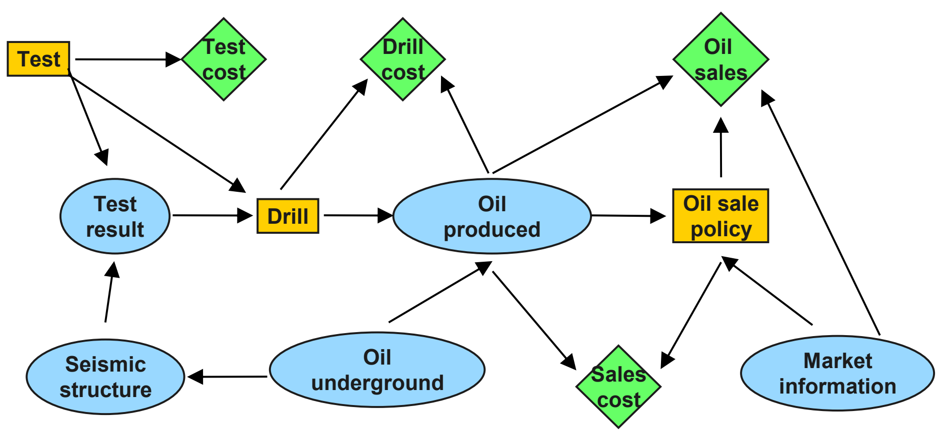
Graphical Models

- Describe structure in large problems
  - Large complex system $f(X)$
  - Made of "smaller", "local" interactions $f_\alpha(X_\alpha)$
  - Complexity emerges through interdependence
- Examples & Tasks
  - Mixed inference (marginal MAP, MEU, …)

$$f(\mathbf{x}_M^*) = \max_{\mathbf{x}_M} \sum_{\mathbf{x}_S} \prod_\alpha f_\alpha(\mathbf{x}_\alpha)$$

Influence diagrams &
optimal decision-making

(the "oil wildcatter" problem)
e.g., [Raiffa 1968; Shachter 1986]



Inference Queries/Tasks

- Maximum A Posteriori (MAP)

$$\max_X \prod_{\alpha \in F} f_\alpha(X_\alpha)$$

NP-hard in general

- The Partition Function

$$Z = \sum_X \prod_{\alpha \in F} f_\alpha(X_\alpha)$$

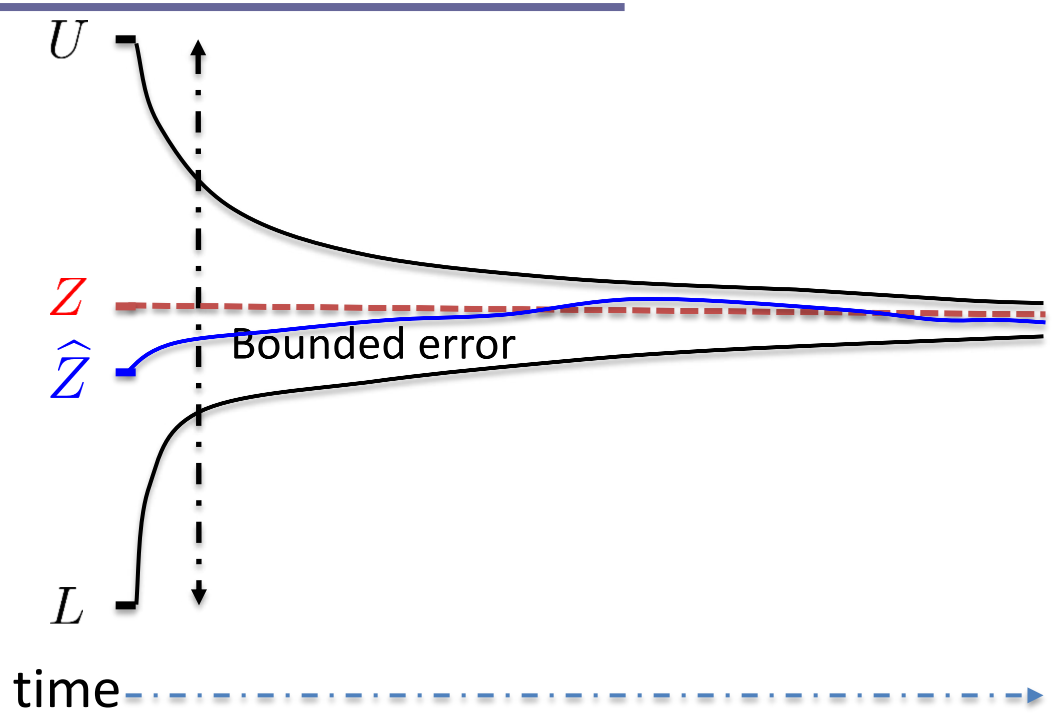
#P-complete [Valiant 1979])

- Marginal MAP (MMAP)

$$\max_{X \setminus X_S} \sum_{X_S} \prod_{\alpha \in F} f_\alpha(X_\alpha)$$

NP^{PP}(decision version) [Park 2002])

Desired Properties: Guarantee, Anytime, Anyspace



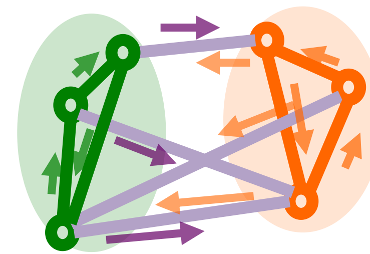
- Anytime
  - valid solution at any point
  - solution quality improves with additional computation
- Anyspace
  - run with limited memory resources

Approximate inference

- Three major paradigms

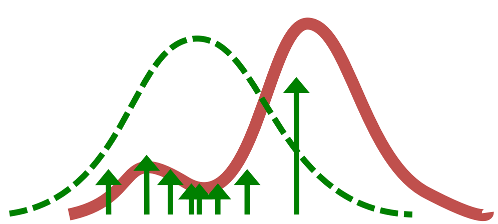
Variational methods

Reason over small subsets of variables at a time



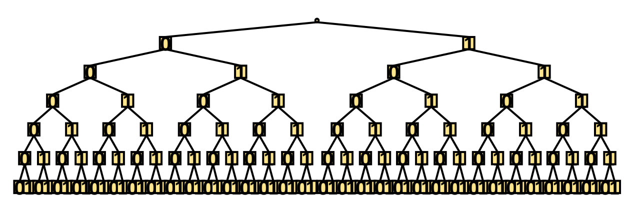
Sampling

Use randomization to estimate averages over the state space



Search

Structured enumeration over all possible states

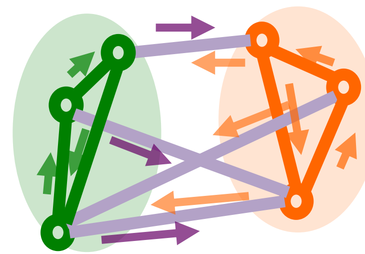


Approximate inference

- Three major paradigms
  - Variational methods (e.g., tree-reweighted belief propagation [Wainwright et al. 2003]), mini-bucket elimination [Dechter & Rish] 2001).

Variational methods

Reason over small subsets of variables at a time

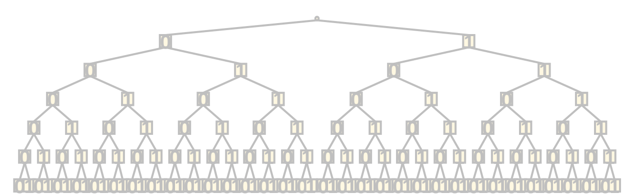


Sampling

Use randomization to estimate averages over the state space

Search

Structured enumeration over all possible states



Approximate inference

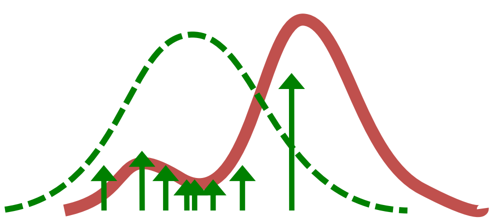
- Three major paradigms
  - (Monte Carlo) Sampling (e.g., importance sampling based (e.g., [Bidyuk & Dechter 2007]), approximate hash-based counting (e.g., [Chakraborty et al. 2016])).

Variational methods

Reason over small subsets of variables at a time

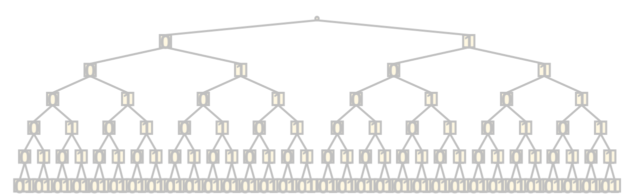
Sampling

Use randomization to estimate averages over the state space



Search

Structured enumeration over all possible states



Approximate inference

- Three major paradigms
  - (Heuristic) Search (e.g., [Lou et al. 2017], [Viricel et al. 2016], [Henrion 1991]).

Variational methods

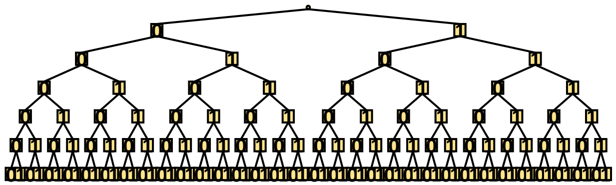
Reason over small subsets of variables at a time

Sampling

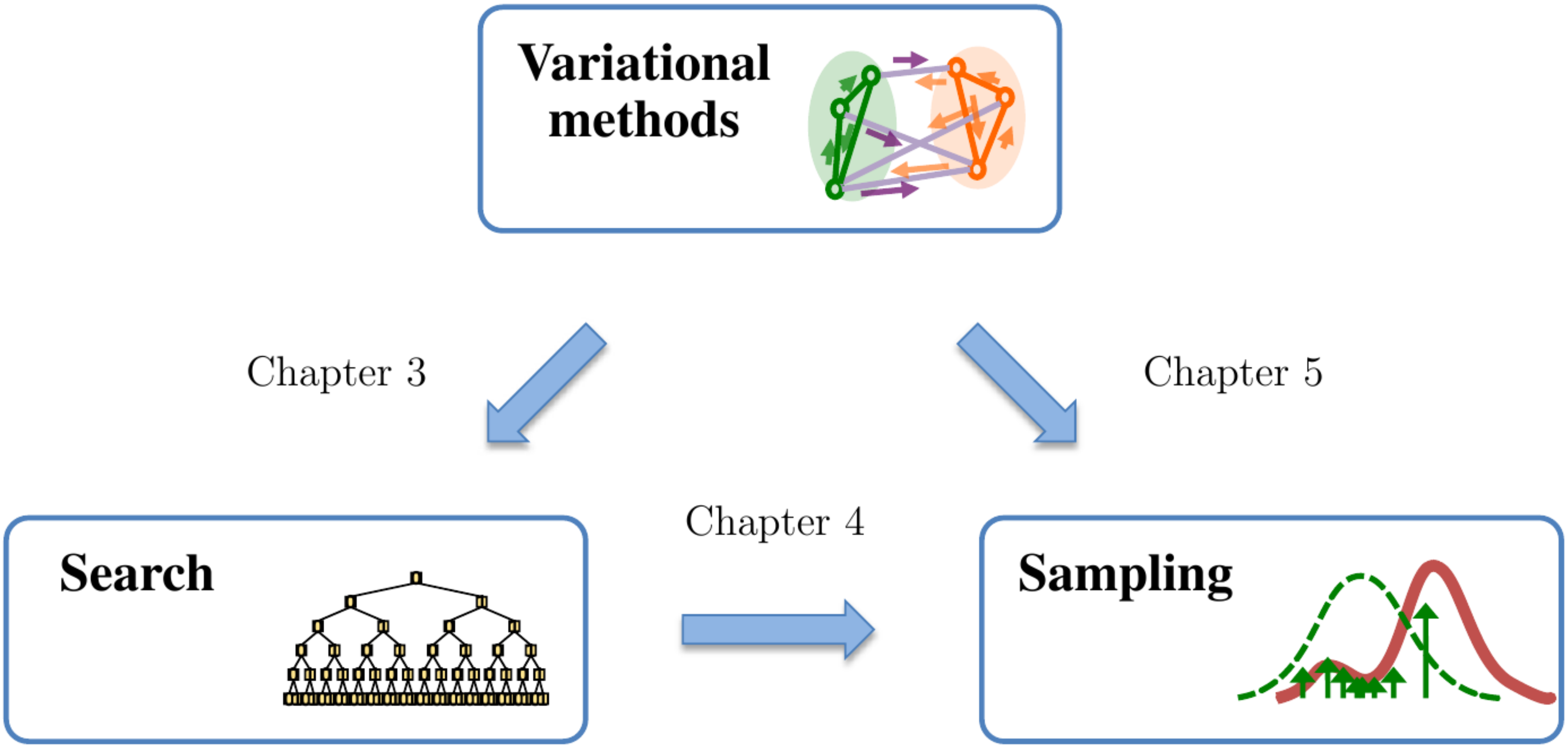
Use randomization to estimate averages over the state space

Search

Structured enumeration over all possible states

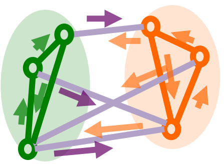


Main Contributions of This Thesis



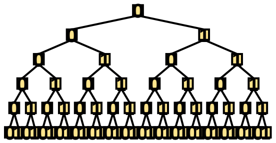
Chapter 3: Best-first Search Aided by Variational Heuristics

Variational methods



provide pre-compiled heuristics

Search

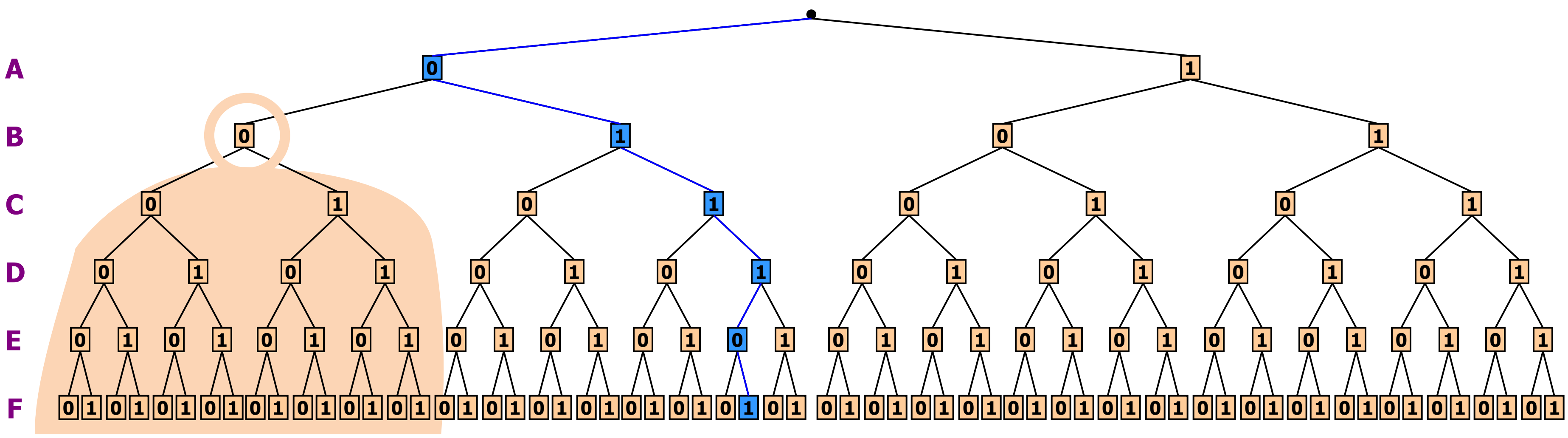


AND/OR best-first search (AOBFS)

unified best-first search (UBFS)

Search Trees and Summation

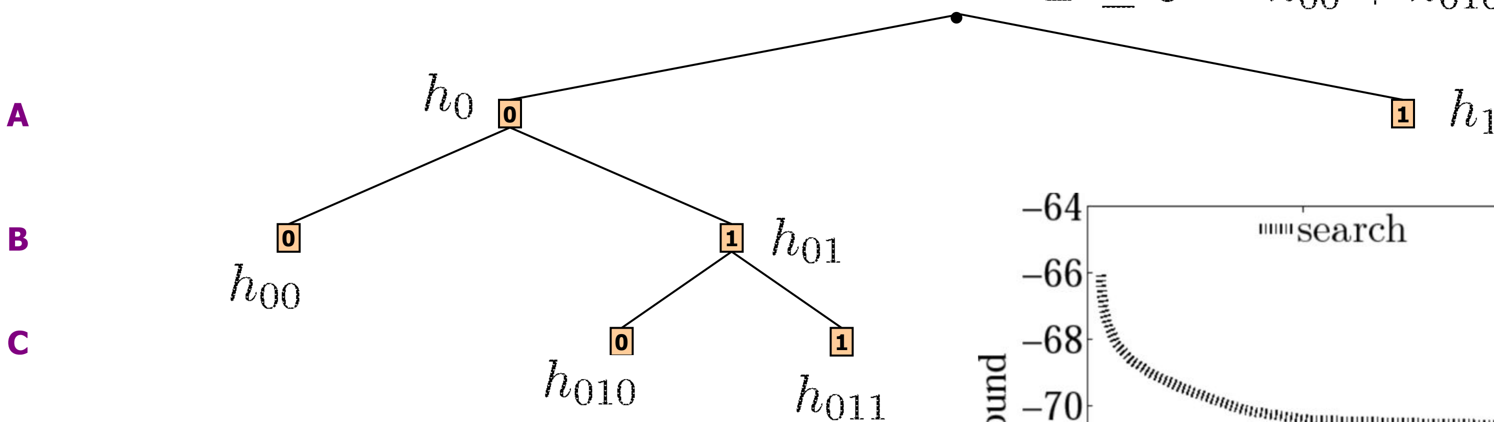
- Organize / structure the state space
  - Leaf nodes = model configurations
  - “Value” of a node = sum of configurations below

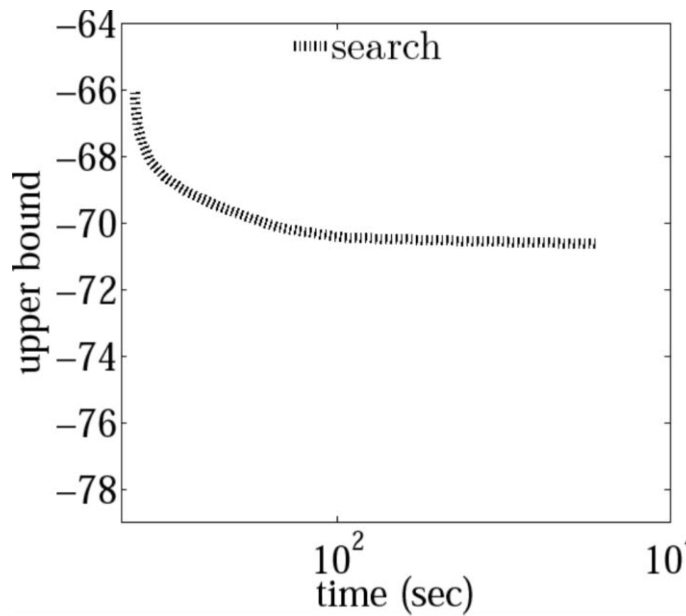


Search Trees and Summation

- Heuristic search for summation
  - Heuristic function upper bounds value (sum below) at any node
  - Expand tree and compute updated bounds

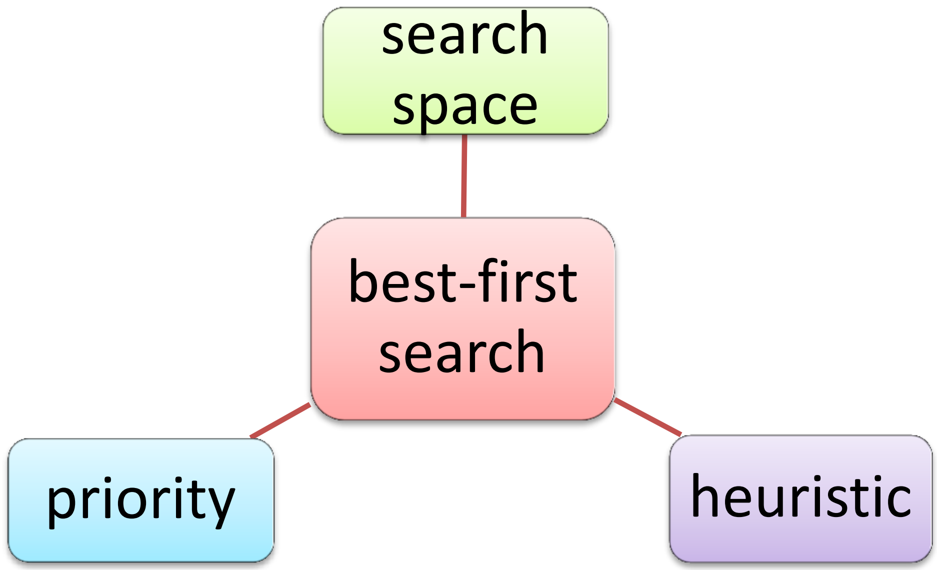
$$Z \leq U = h_{00} + h_{010} + h_{011} + h_1$$





AND/OR Best-first Search (AOBFS)

AND/OR search tree

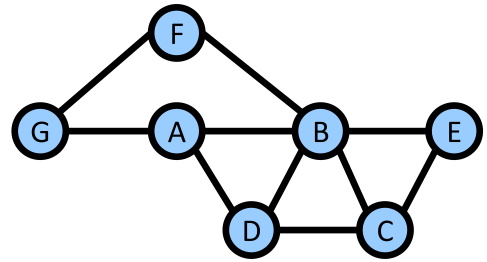


potentially reduce the bound gap $U - L$ on Z most

weighted mini-bucket

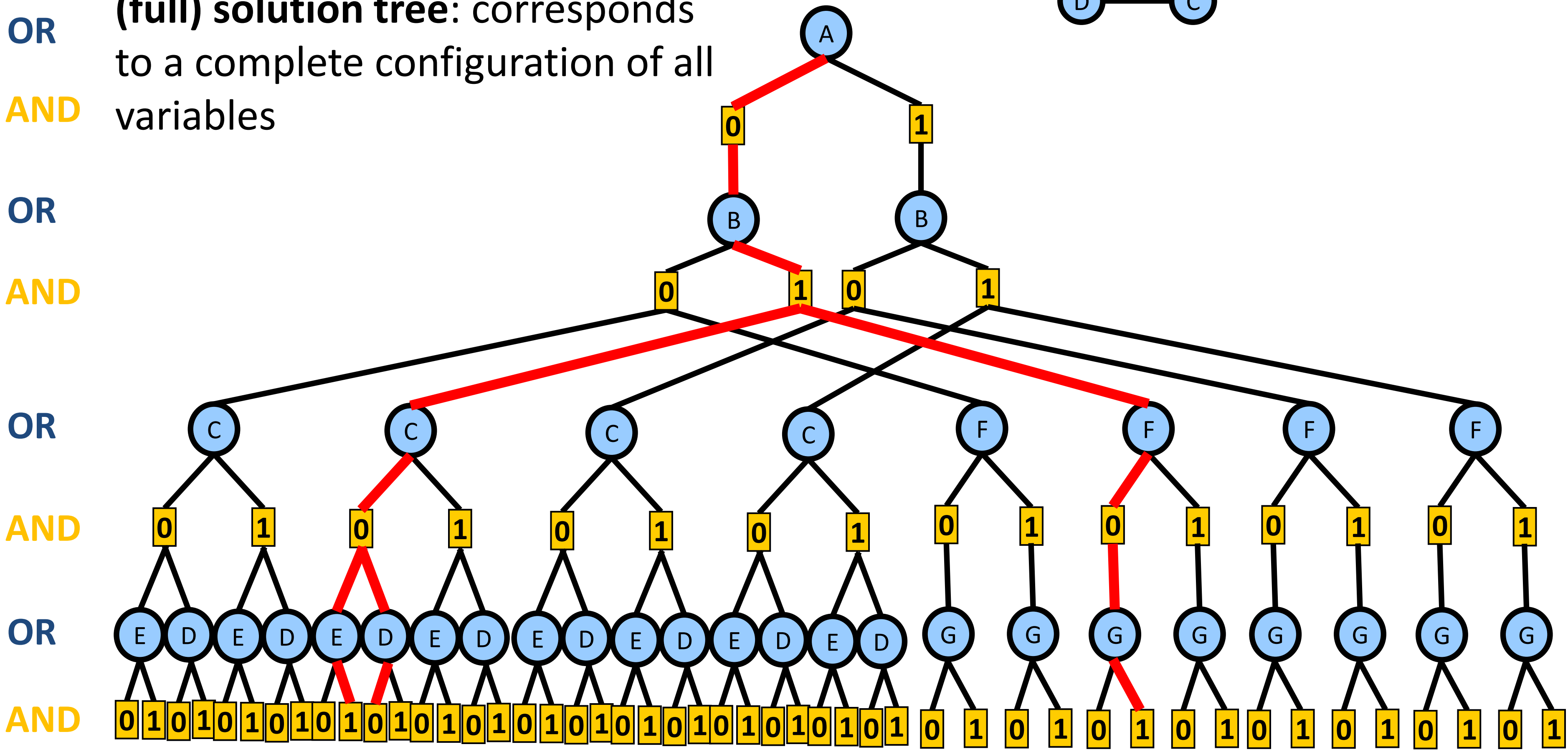
AND/OR Search Trees

[Nillson 1980, Dechter and Mateescu 2007]



$\text{search space size } O(d^h)$

(full) solution tree: corresponds to a complete configuration of all variables

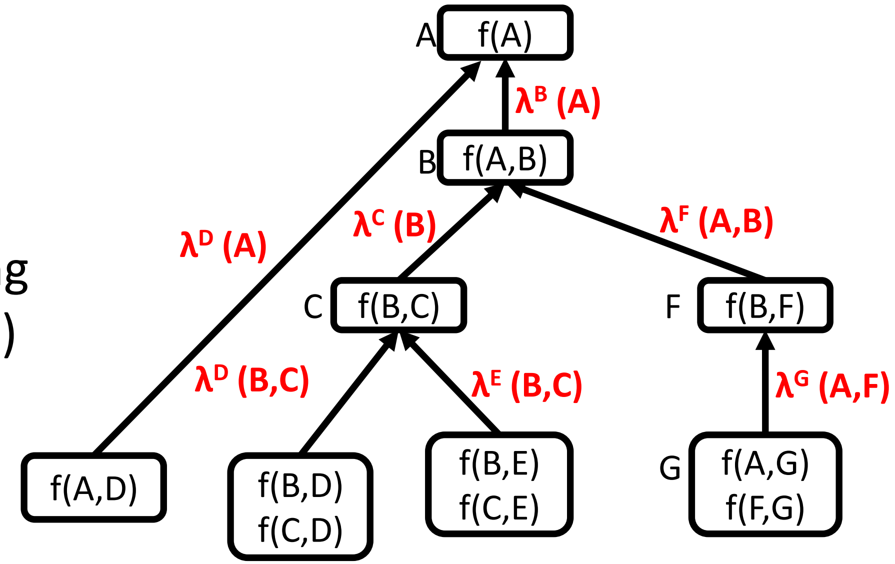


weighted mini-bucket (WMB) Heuristics

[Liu and Ihler, ICML'11]

$$h(A) = \lambda^D(A)\lambda^B(A)$$
$$h(A,B) = \lambda^D(A)\lambda^C(B)\lambda^F(A,B)$$
$$h(A,B,C) = \lambda^D(A)\lambda^D(B,C)\lambda^E(B,C)\lambda^F(A,B)$$
$$\vdots$$
$$h(\mathbf{X}) = 1$$

- ❑ Formed by intermediately generated factors (called messages, e.g.,**λ^D (A)**)
- ❑ Upper (or lower) bound of the node value.
- ❑ Monotonic: Resolving relaxations using search makes heuristics more (no less) accurate.
- ❑ Quality can be roughly controlled by the *ibound*.



$$\text{ibound} = 2$$

Priority

- Intuition: expand the frontier node that potentially reduces the bound gap $U - L$ ($L <= Z <= U$) most

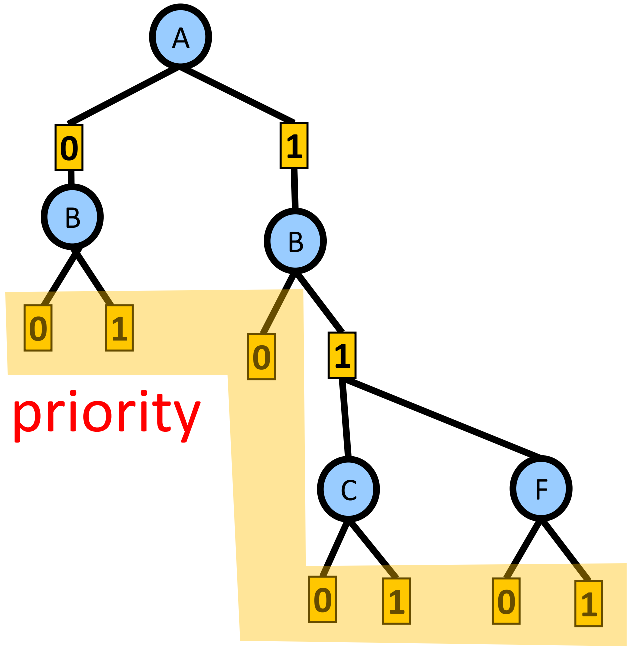
$$gap(n) := U(n) - L(n)$$ ← gap priority

where

$$U(n) := g(n)h^+(n) \prod_{s \in br(n)} h^+(s)$$ ← upper priority

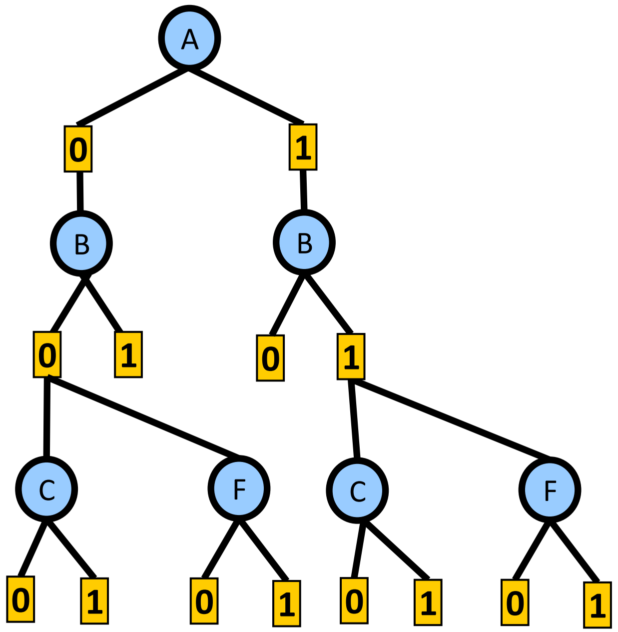
$$L(n) := g(n)h^-(n) \prod_{s \in br(n)} h^-(s)$$

$br(n)$: set of OR nodes adjacent to some node on the path from the root to n

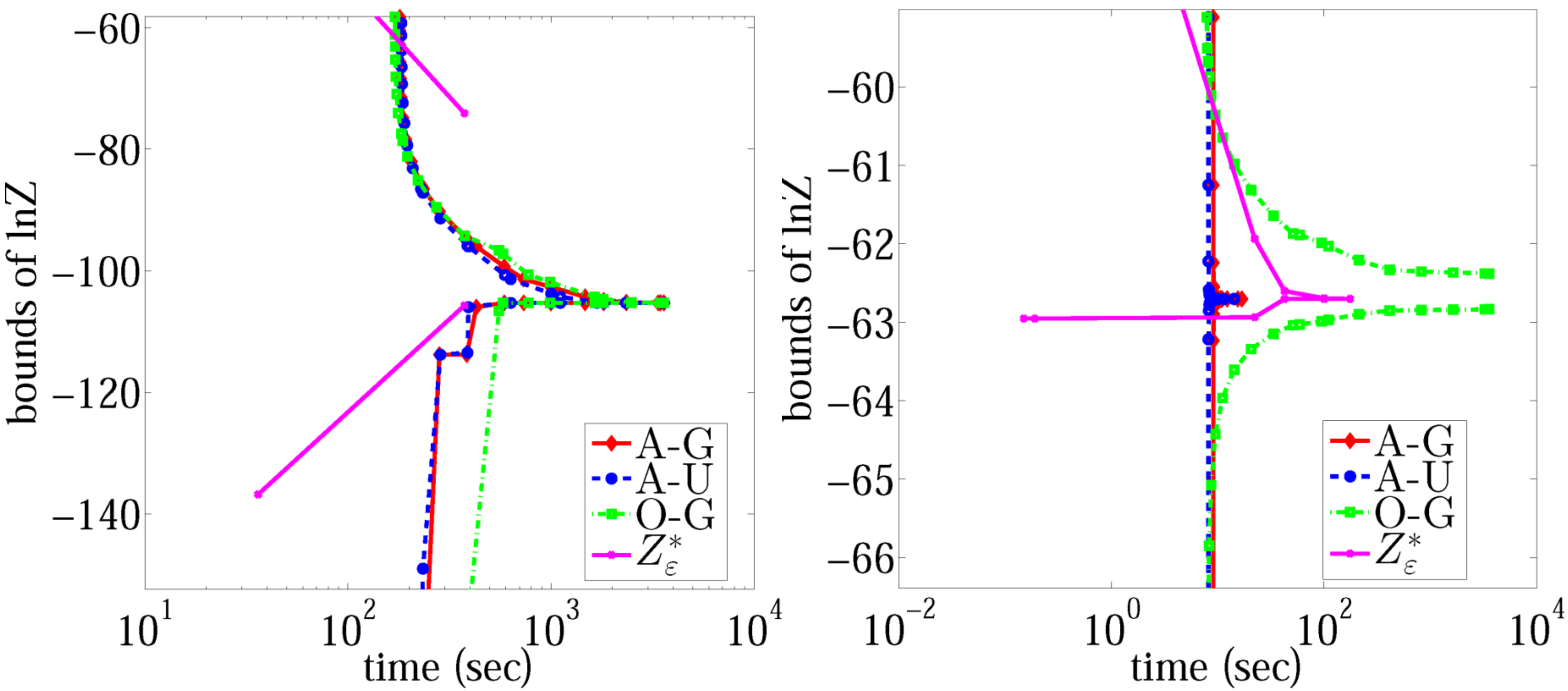


Overcome The Memory Limit

- Main strategy (SMA*-like [Russell 1992])
  - Keep track of the lowest-priority node as well
  - When reach the memory limit, delete the lowest-priority nodes, and keep expanding the top-priority ones



Anytime Behavior of AOBFS



(a) PIC'11/queen5_5_4 (b) Protein/1g6x

Aggregated Results

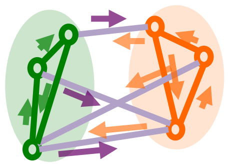
- Number of instances solved to "*tight*" tolerance interval. The best (most solved) for each setting is **bolded**.

"Tight": $\log U - \log L < 10^{-3}$

	PIC'11 (23)	Protein (50)	BN (50)	CPD (100)
Memory: 1GB/4GB/16GB				
A-G	**18/18/19**	**16/17/19**	32/**40**/**42**	95/98/**100**
A-U	**18/18/19**	15/**17**/**19**	32/**40**/41	93/95/**100**
O-G	16/**18**/**19**	9/12/13	28/36/38	95/98/**100**
Z^*_ε	13/13/15	12/12/12	30/31/31	**100/100/100**
VEC	12/14/**19**	14/15/15	**36**/38/39	36/52/56
M-D	14/14/14	9/9/11	23/23/24	7/7/8

Best-first Search Aided by Variational Heuristics

Variational methods

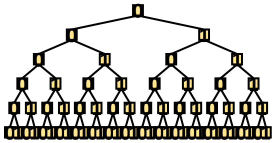


weighted mini-bucket (WMB)

[Liu and Ihler, ICML'11]

provide optimized heuristics

Search



AND/OR best-first search (AOBFS) for Z

unified best-first search (UBFS) for marginal MAP

Unified Best-first Search (UBFS)

- Idea: unify max- and sum- inference in one search framework
  - avoids some unnecessary exact evaluation of conditional summation problems
- Principle: focus on reducing the upper bound of MMAP as quickly as possible
- How it works:
  - Track the current most promising (partial) MAP configuration, i.e., one with the highest upper bound
  - Expand the most "influential" frontier node of that (partial) MAP configuration
    - Frontier node that contributes most to its upper bound
    - Identified by a specially designed "double-priority" system

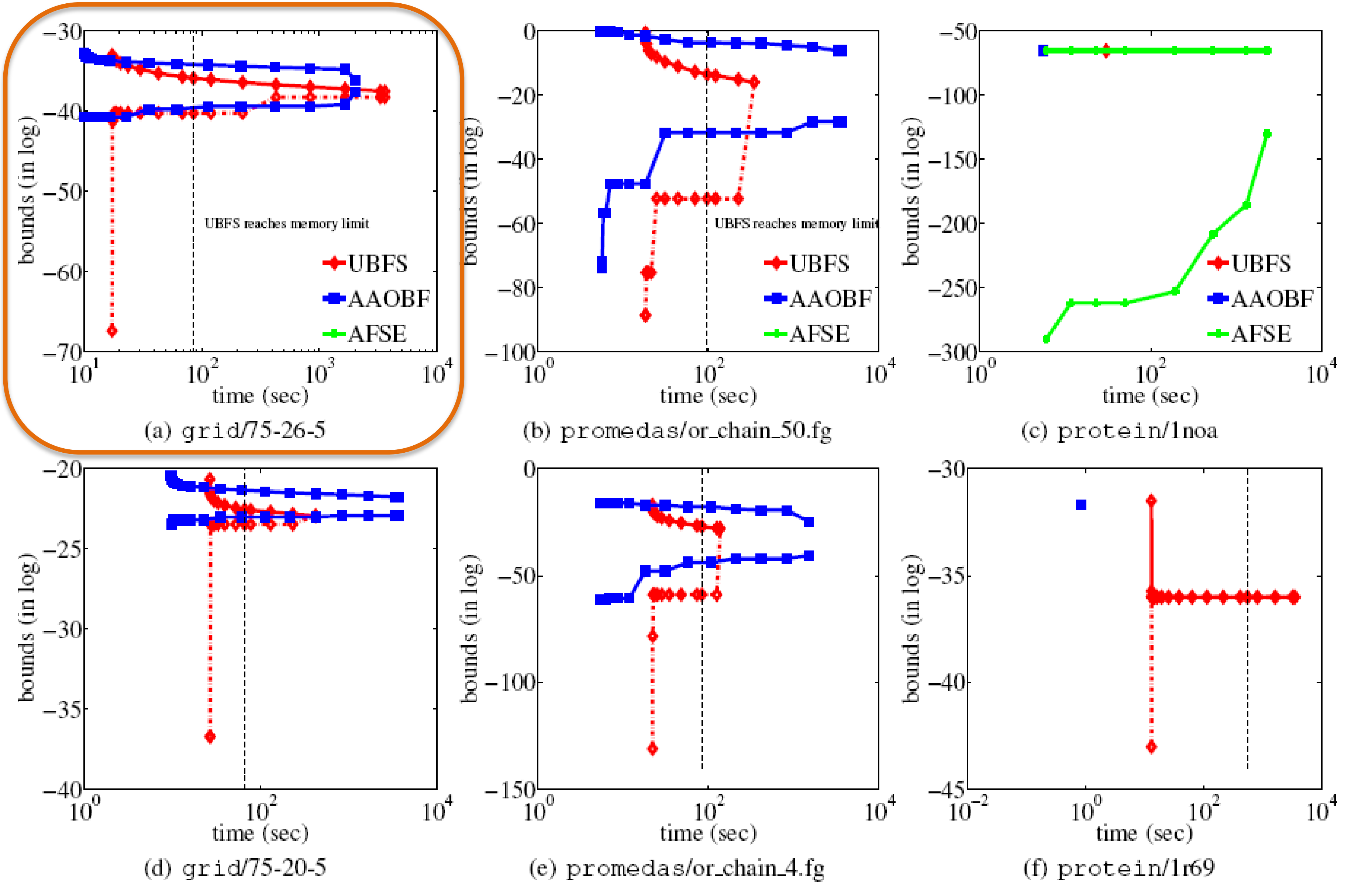


Figure 2: Anytime bounds for two instances per benchmark, with 50% MAX variables. AFSE is missing if it ran out of memory before producing bounds; XOR_MMAP failed to produce bounds on these instances. UBFS lower bounds are computed offline and shown only for reference. Black dotted lines mark UBFS reaching the 4GB memory limit; time budget 1 hour.

Table 3: Number of instances that an algorithm achieves the best upper bounds at each timestamp (1 min, 10 min, and 1 hour) for each benchmark. 50% MAX variables. The best for each setting is bolded.

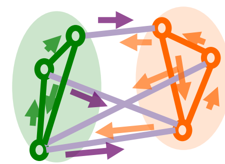
	grid	promedas	protein
# instances	100	100	50
Timestamp: 1min/10min/1hr			
UBFS	**85/84/89**	**83/86/87**	**46/50/50**
AAOBF	33/46/44	47/47/47	17/16/18
AFSE	0/0/0	0/0/0	5/5/5

Table 4: Number of instances that an algorithm achieves best upper bounds at each given timestamp (1 min, 10 min, and 1 hour) for each benchmark. 10% MAX variables. The best for each setting is bolded.

	grid	promedas	protein
# instances	100	100	50
	Timestamp: 1min/10min/1hr		
UBFS	**99/100/100**	**88/99/99**	**43/50/50**
AAOBF	1/1/3	17/12/17	15/9/9
AFSE	0/0/0	10/8/10	7/7/7

Chapter 4: Sampling Enhanced by Best-first Search

Variational methods



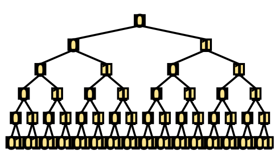
weighted mini-bucket (WMB)

provide heuristic

provide proposal

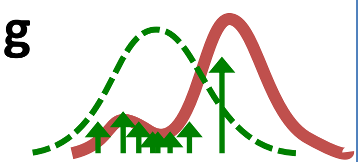
WMB-IS [Liu, Fisher, Ihler, NIPS'15]

Search



refine proposal

Sampling



AND/OR best-first search (AOBFS)

dynamic importance sampling (DIS)

mixed dynamic importance sampling (MDIS)

Monte Carlo Estimators

- Most basic form: empirical estimate of probability

$$\mathbb{E}[u(x)] = \int p(x)u(x) \approx \widehat{u} = \frac{1}{m}\sum_i u(\tilde{x}^{(i)}), \quad \tilde{x}^{(i)} \sim p(x)$$

- Relevant considerations
  - Able to sample from the target distribution p(x)?
  - Able to evaluate p(x) explicitly, or only up to a constant?

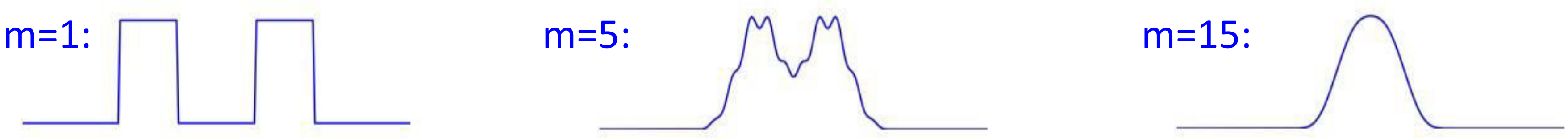
- "Anytime" properties
  - Unbiased estimator, $\mathbb{E}[\widehat{u}] = \mathbb{E}[u(x)]$
    or asymptotically unbiased, $\mathbb{E}[\widehat{u}] \to \mathbb{E}[u(x)]$ as $m \to \infty$
  - Variance of the estimator decreases with m

Monte Carlo Estimators

- Most basic form: empirical estimate of probability

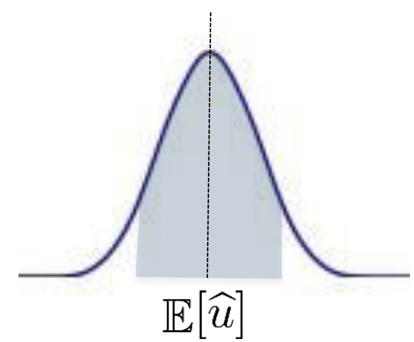
$$\mathbb{E}[u(x)] = \int p(x)u(x) \approx \widehat{u} = \frac{1}{m}\sum_i u(\tilde{x}^{(i)}), \quad \tilde{x}^{(i)} \sim p(x)$$

- Central limit theorem
  - $\widehat{u}$ is asymptotically Gaussian:



- Finite-sample confidence intervals
  - If $u(x)$ is bounded, e.g., $u(x^{(i)}) \in [0, 1]$, probability concentrates rapidly around the expectation:

$$\Pr\left[|\widehat{u} - \mathbb{E}[\widehat{u}]| > \epsilon\right] \leq O(e^{-m\epsilon^2})$$



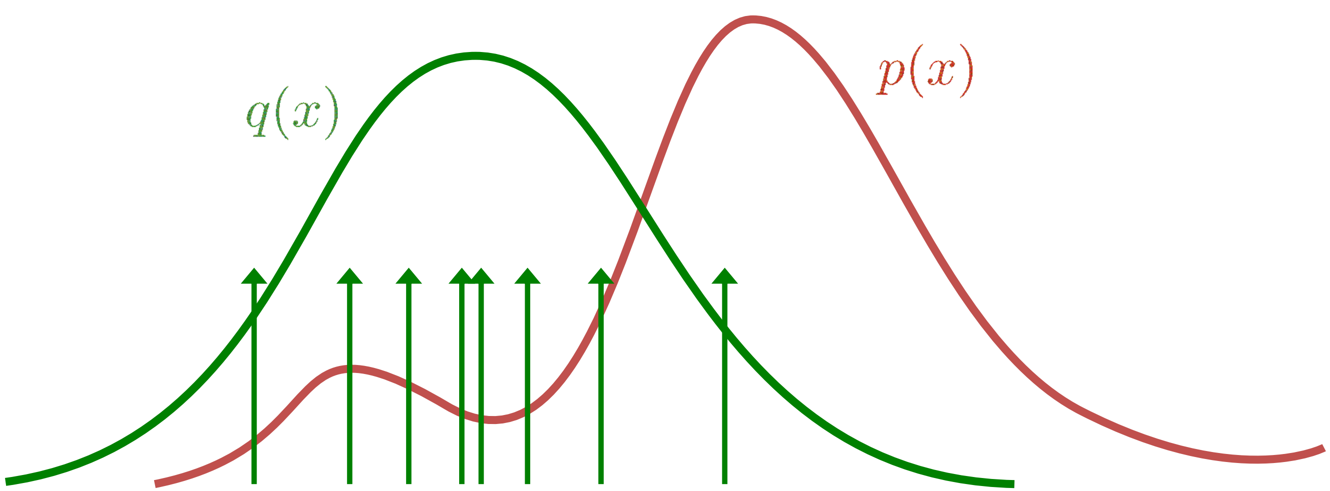
Importance Sampling

- Basic empirical estimate of probability:

$$\mathbb{E}[u(x)] = \int p(x)u(x) \approx \widehat{u} = \frac{1}{m}\sum_i u(\tilde{x}^{(i)}), \quad \tilde{x}^{(i)} \sim p(x)$$

- Importance sampling:

$$\int p(x)u(x) = \int q(x)\frac{p(x)}{q(x)}u(x) \approx \frac{1}{m}\sum_i \frac{p(\tilde{x}^{(i)})}{q(\tilde{x}^{(i)})}u(\tilde{x}^{(i)}) \qquad \tilde{x}^{(i)} \sim q(x)$$



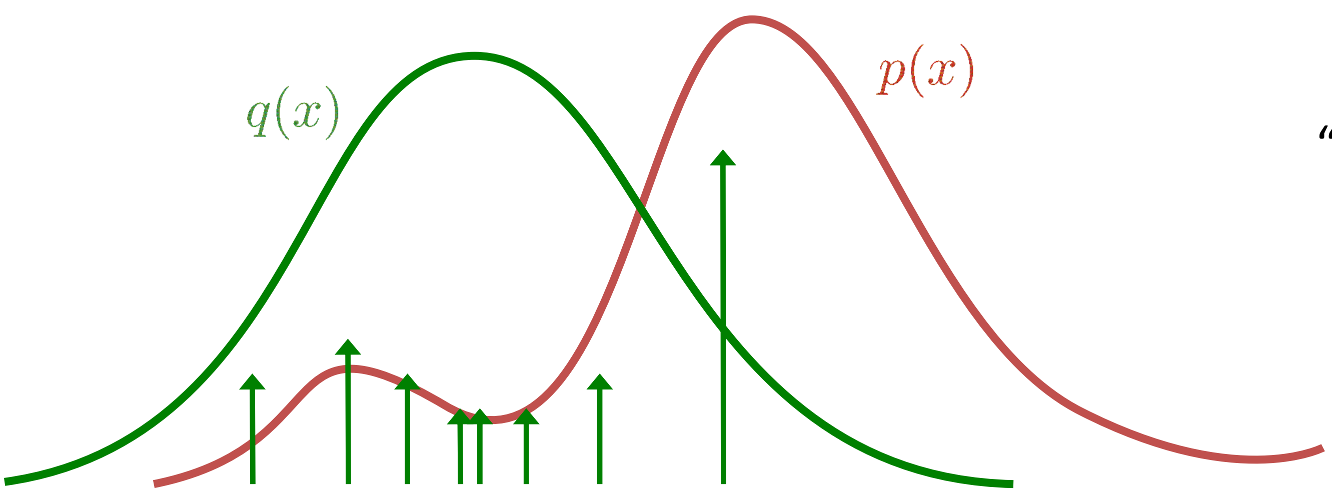
Importance Sampling

- Basic empirical estimate of probability:

$$\mathbb{E}[u(x)] = \int p(x)u(x) \approx \hat{u} = \frac{1}{m}\sum_i u(\tilde{x}^{(i)}), \quad \tilde{x}^{(i)} \sim p(x)$$

- Importance sampling:

$$\int p(x)u(x) = \int q(x)\frac{p(x)}{q(x)}u(x) \approx \frac{1}{m}\sum_i \frac{p(\tilde{x}^{(i)})}{q(\tilde{x}^{(i)})}u(\tilde{x}^{(i)}) \qquad \tilde{x}^{(i)} \sim q(x)$$



"importance weights"

$$w^{(i)} = \frac{p(\tilde{x}^{(i)})}{q(\tilde{x}^{(i)})}$$

Choosing a proposal

[Liu, Fisher, Ihler, NIPS'15]

- Can use WMB upper bound to define a proposal $q_{\mathrm{wmb}}(x)$

$$\tilde{\mathbf{b}} \sim w_1\, q_1(b|\tilde{a}, \tilde{c}) \;+\; w_2\, q_2(b|\tilde{d}, \tilde{e})$$

Weighted mixture:

use mini-bucket 1 with probability w_1

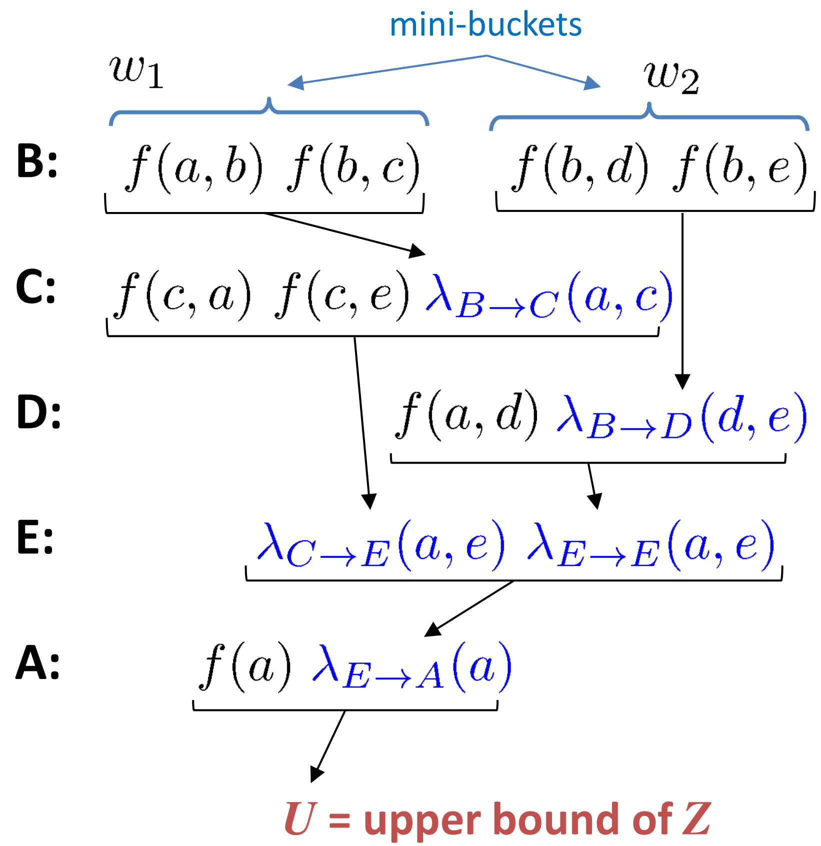
or, mini-bucket 2 with probability $w_2 = 1 - w_1$

where

$$q_1(b|a,c) = \left[\frac{f(a,b)\cdot f(b,c)}{\lambda_{B\to C}(a,c)}\right]^{\frac{1}{w_1}}$$

⋮

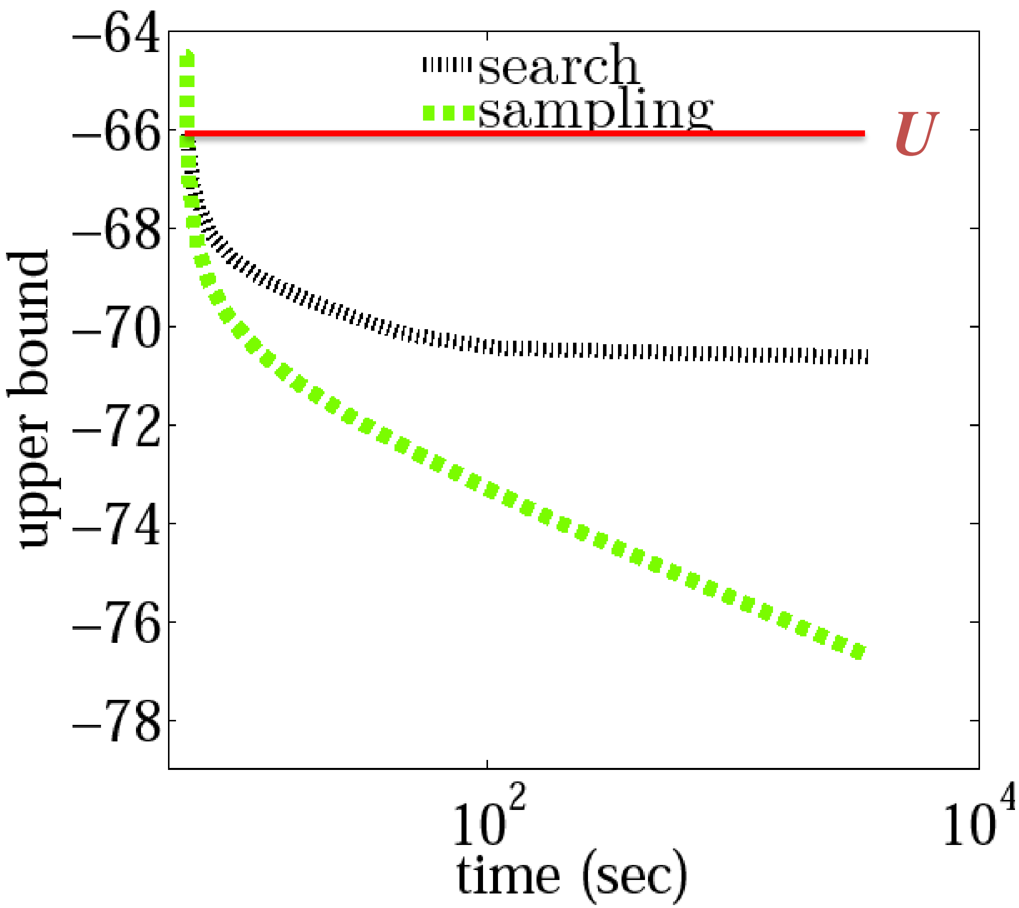
$$\tilde{\mathbf{a}} \sim q(A) = f(a) \cdot \lambda_{E\to A}(a)/U$$



Key insight: provides bounded importance weights!

$$0 \le f(x)/q_{\mathrm{wmb}}(x) \le U \quad \forall x$$

WMB-IS

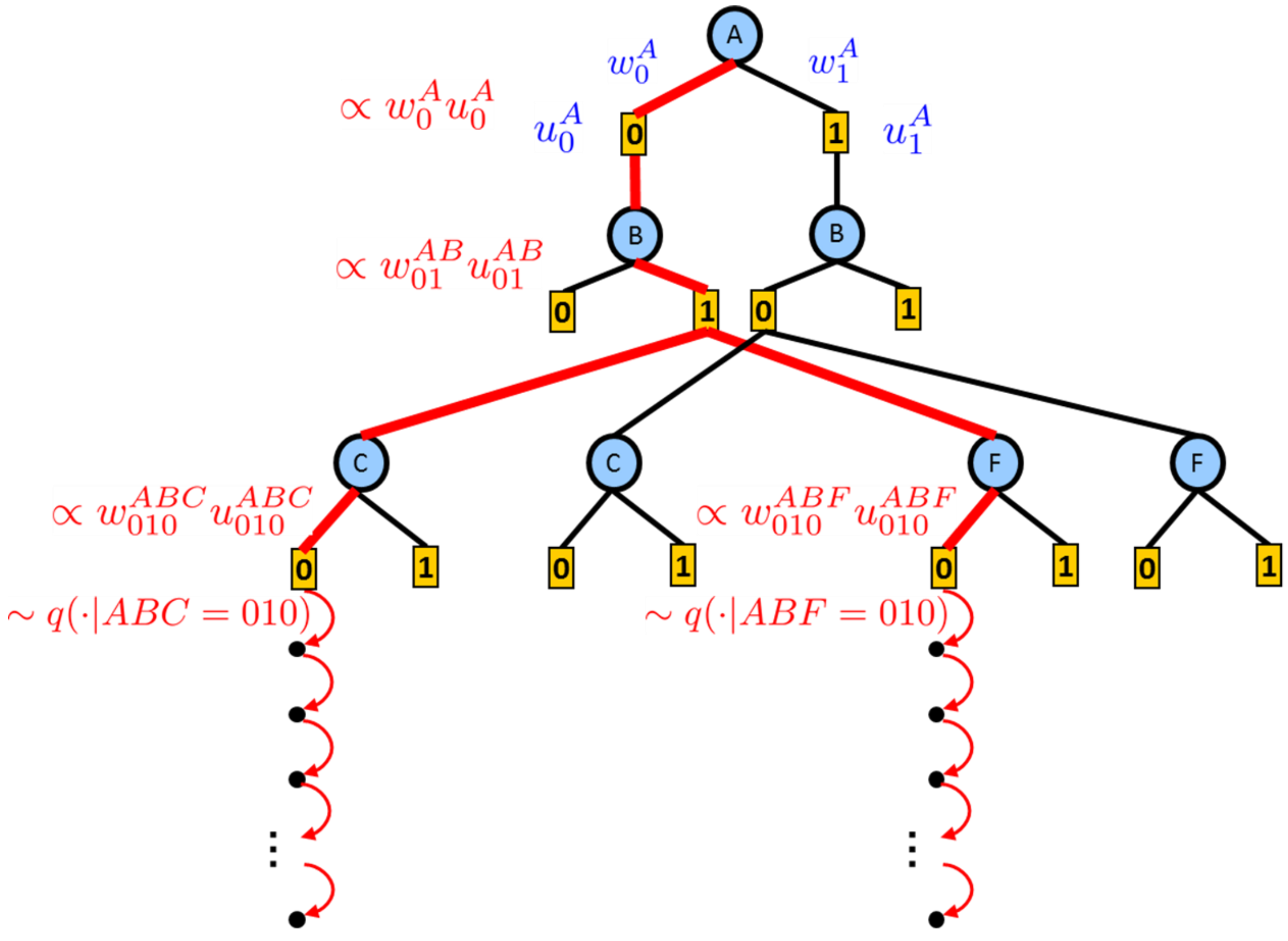


$$\Pr\left[|\hat{Z} - Z| > \epsilon\right] \le 1 - \delta$$

$$\epsilon = \sqrt{\frac{2\hat{V}\log(4/\delta)}{m}} + \frac{7\,U\,\log(4/\delta)}{3(m-1)}$$

"Empirical Bernstein" bounds

Two-step Sampling



Boundedness of Two-step Sampling

Proposition:

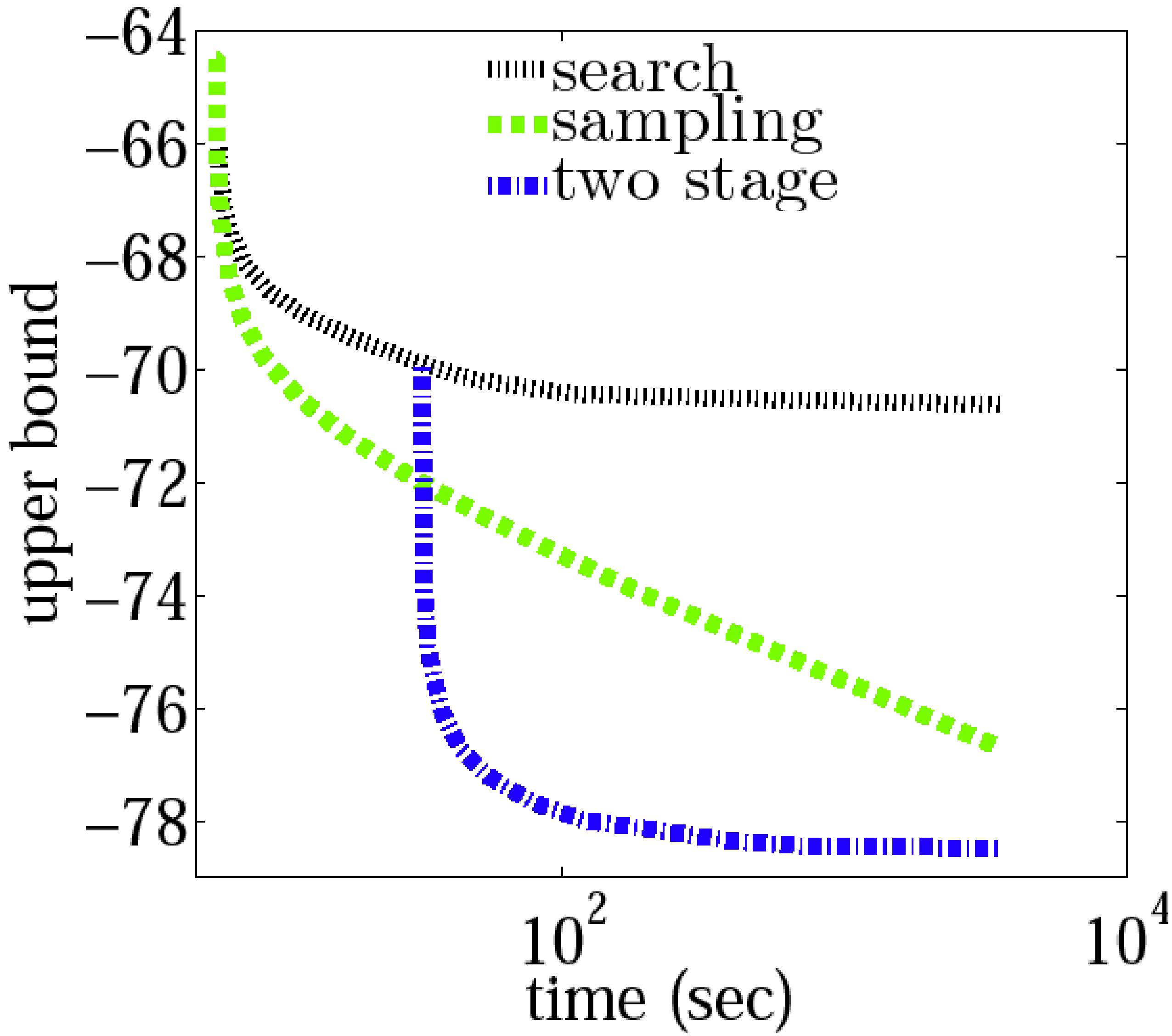
$$f(x)/q^{\mathcal{S}}(x) \leq U^{\mathcal{S}}, \qquad \mathbb{E}\left[f(x)/q^{\mathcal{S}}(x)\right] = Z$$

$\mathcal{S}$: current search tree

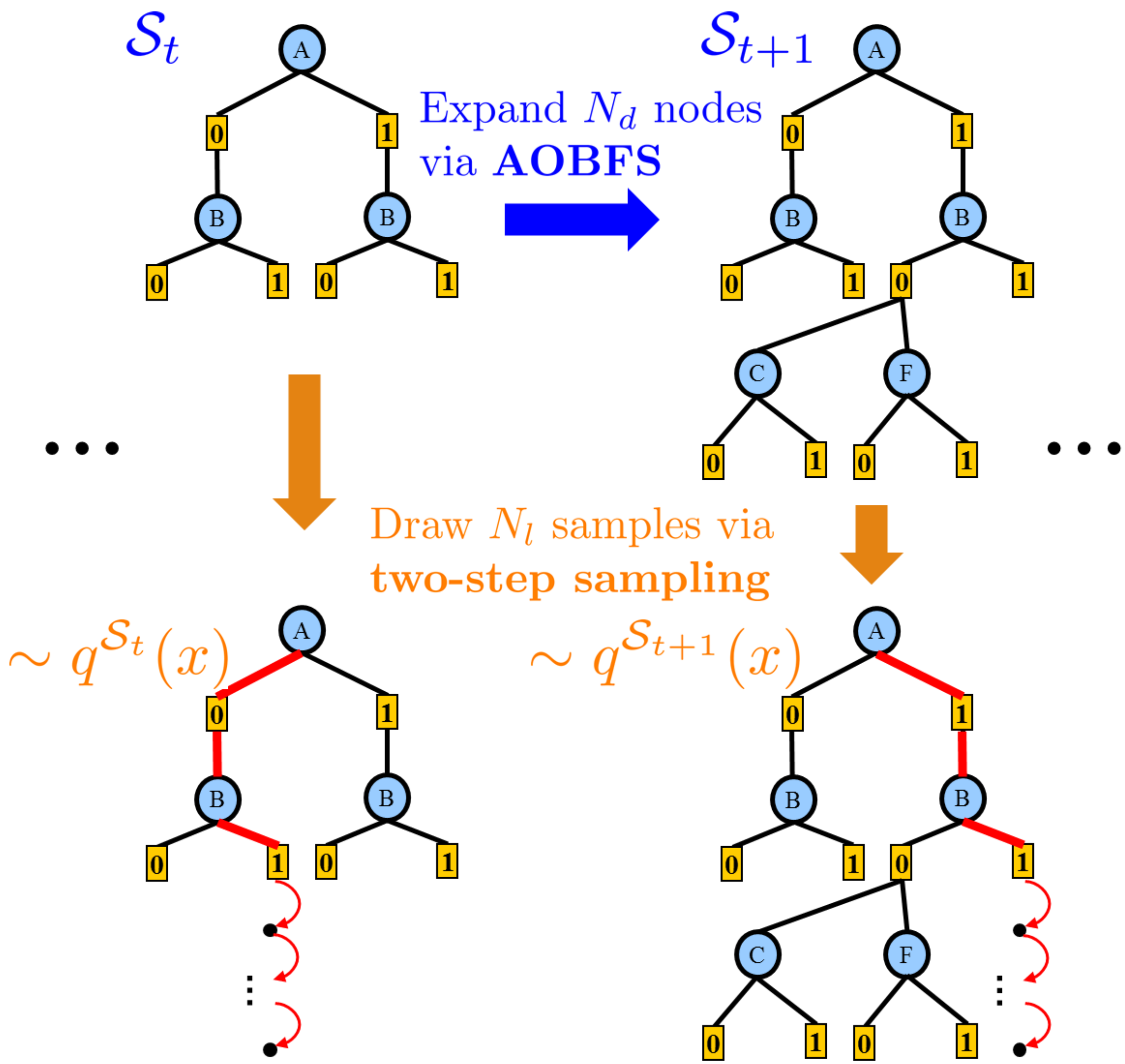
$U^{\mathcal{S}}$: refined upper bound by current search tree

$q^{\mathcal{S}}(x)$: proposal distribution defined by two-step sampling

Two Stage Sampling



Dynamic Importance Sampling (DIS)



Sample Aggregation Strategy for DIS

- Weighted average of importance weights: weight each sample with its corresponding upper bound.

$$\widehat{Z} = \frac{\mathrm{HM}(\boldsymbol{U})}{N} \sum_{i=1}^{N} \frac{\widehat{Z}_i}{U_i}, \quad \mathrm{HM}(\boldsymbol{U}) = \left[\frac{1}{N} \sum_{i=1}^{N} \frac{1}{U_i} \right]^{-1}$$

$\widehat{Z}_i$: importance weight corresponding to the i-th sample

U_i : upper bound being refined in the search process

$$\widehat{Z} \leq \mathrm{HM}(\boldsymbol{U}) \qquad (\textbf{bounded})$$

$$\mathbb{E}\,\widehat{Z} = Z \qquad (\textbf{unbiased})$$

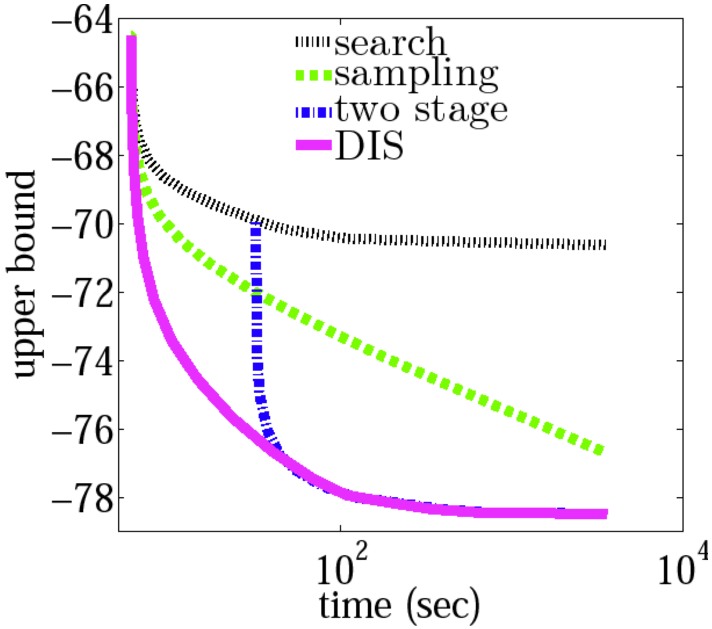
Finite-sample Bounds for DIS

Theorem: Define the deviation term

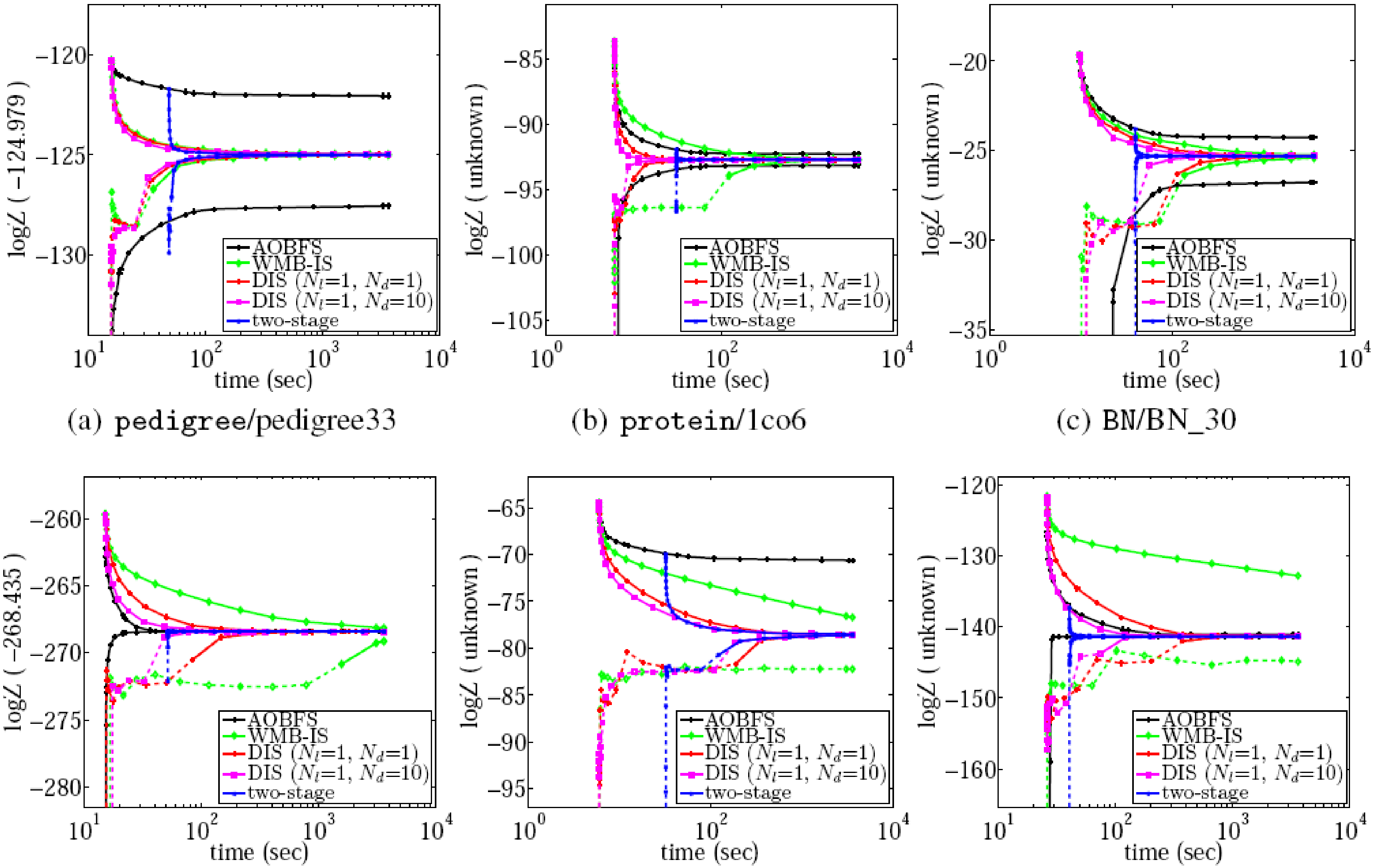
$$\Delta = \mathrm{HM}(\boldsymbol{U})\left(\sqrt{\frac{2\widehat{\mathrm{Var}}(\{\widehat{Z}_i/U_i\}_{i=1}^N)\ln(2/\delta)}{N}} + \frac{7\ln(2/\delta)}{3(N-1)}\right)$$

then, $\Pr[Z \le \widehat{Z} + \Delta] \ge 1 - \delta$ and $\Pr[Z \ge \widehat{Z} - \Delta] \ge 1 - \delta$.

$\widehat{\mathrm{Var}}(\{\widehat{Z}_i/U_i\}_{i=1}^N)$: empirical variance of $\{\widehat{Z}_i/U_i\}_{i=1}^N$.



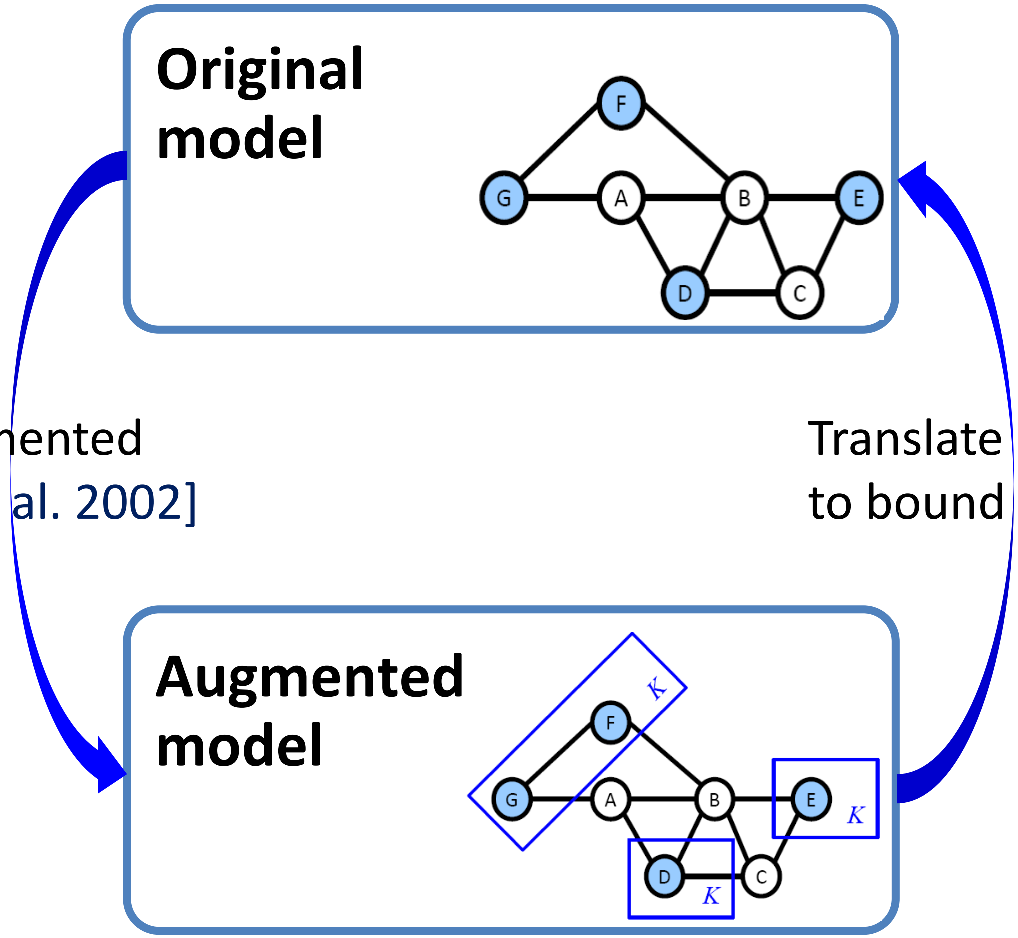
Results on Individual Instances



(a) pedigree/pedigree33 (b) protein/1co6 (c) BN/BN_30

(d) pedigree/pedigree37 (e) protein/1bgc (f) BN/BN_129

Mixed Dynamic Importance Sampling (MDIS)



Construct an augmented model [Doucet et al. 2002]

Translate bounds back to bound MMAP

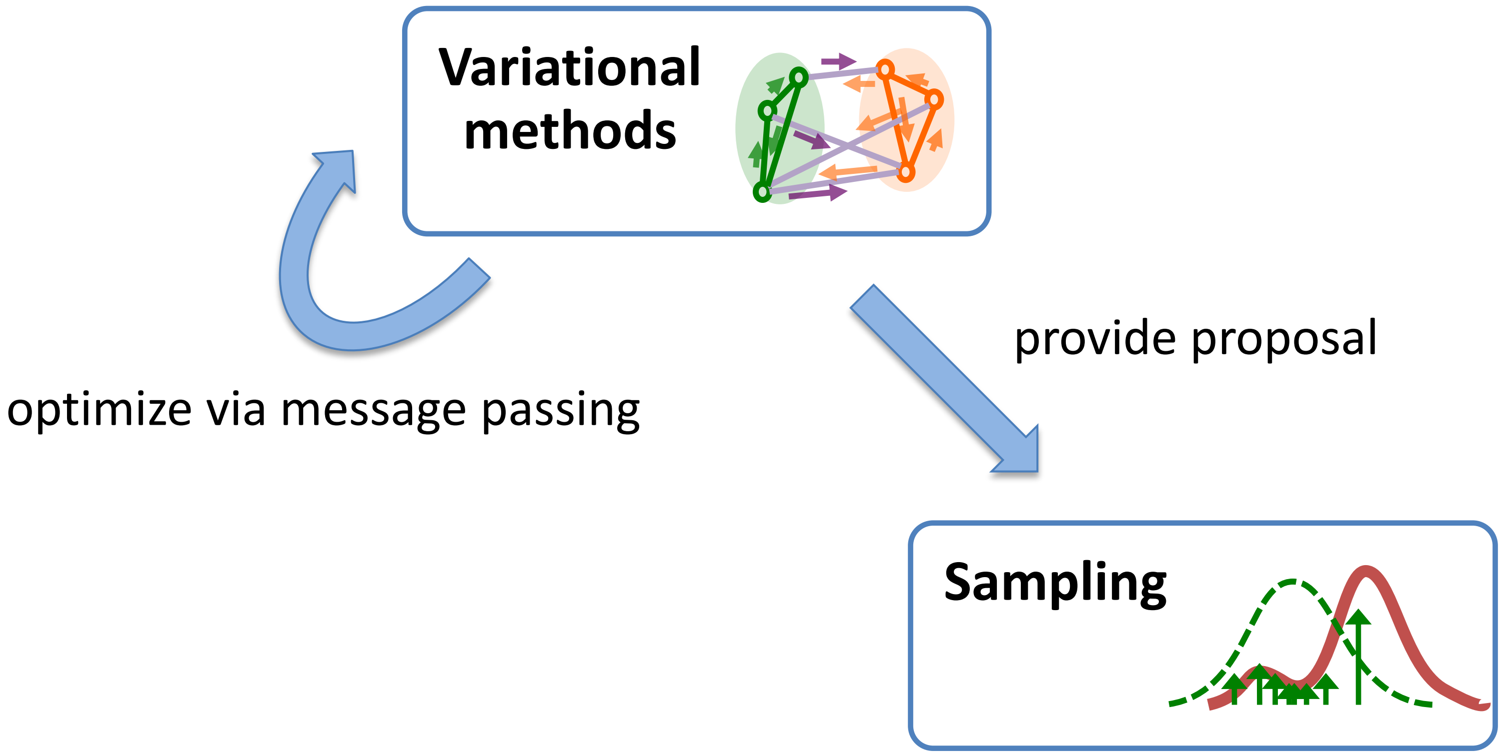
Generalize DIS to provide finite-sample bounds for a series of summation objectives

Empirical Evaluation for MDIS

Number of instances that an algorithm achieves the best ***lower*** (top) and ***upper*** (down) bounds. (Entries for UBFS are blank since UBFS does not provide lower bounds.)

	grid	promedas	protein	planning
# instances	50	50	44	15
Timestamp: 1min/10min/1hr				
MDIS (K=5)	**47/44/45**	**32/34/31**	**31/27/28**	**14/13/13**
MDIS (K=10)	3/2/1	4/5/6	11/13/14	1/2/2
UBFS	-/-/-	-/-/-	-/-/-	-/-/-
AAOBF	0/4/4	16/21/24	2/4/4	0/0/0
Timestamp: 1min/10min/1hr				
MDIS (K=5)	0/0/0	9/12/13	5/9/15	1/1/1
MDIS (K=10)	0/0/0	10/13/14	9/10/13	1/2/3
UBFS	**50/50/50**	**50/50/50**	**36/32/26**	**14/14/13**
AAOBF	0/0/1	2/4/6	2/2/2	1/1/1

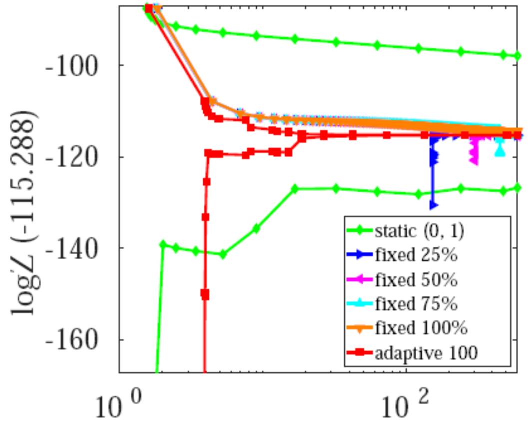
Chapter 5: A General Interleaving Framework



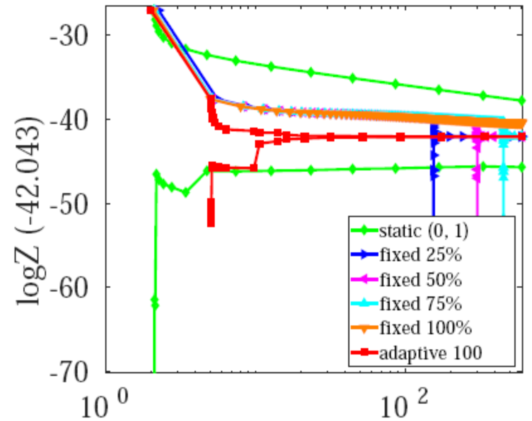
Adaptive Policy

- Idea:
  - Predict unit gains (bound reduction) of a message passing step and a sampling step, respectively.
  - Execute the action with a larger predicted unit gain.

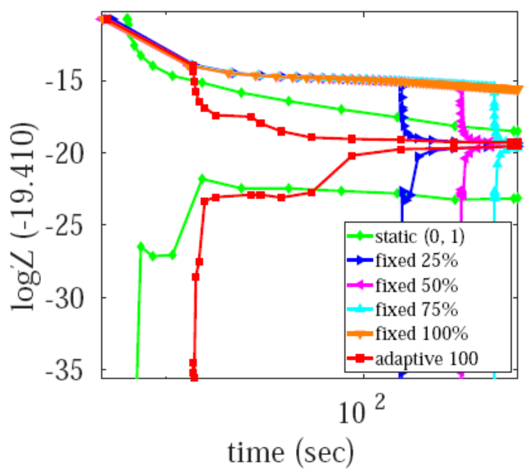
Interleaving v.s. Non-interleaving



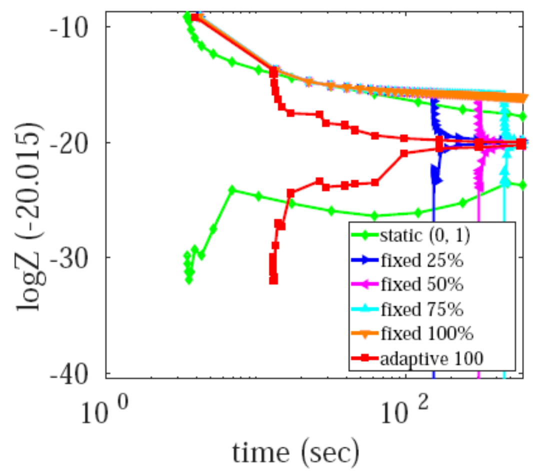
(c) protein/1qsq



(d) protein/1whi

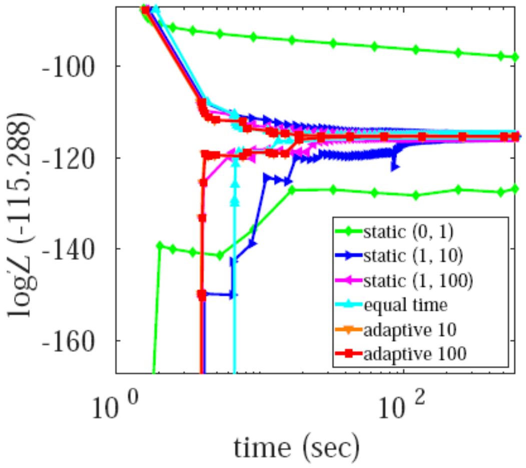


(e) promedas/or_chain_132.fg

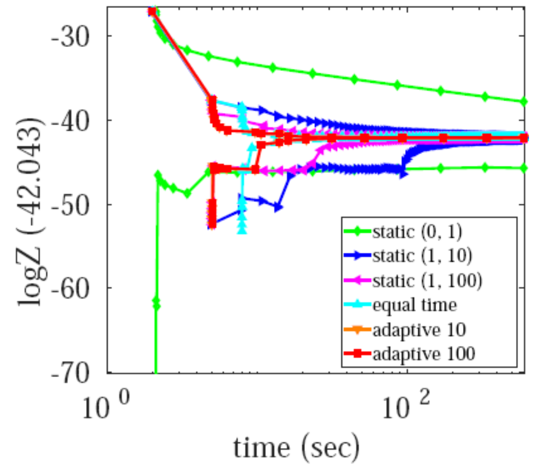


(f) promedas/or_chain_153.fg

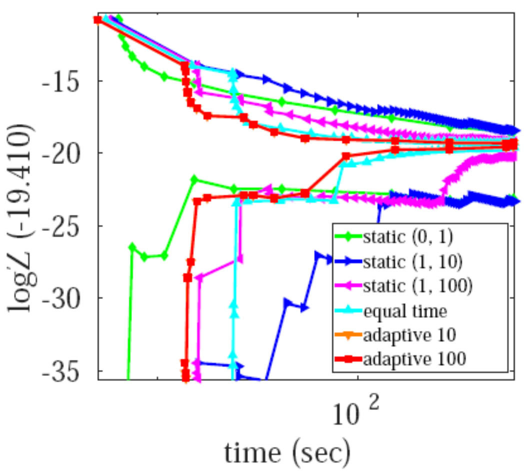
Adaptive v.s. Static



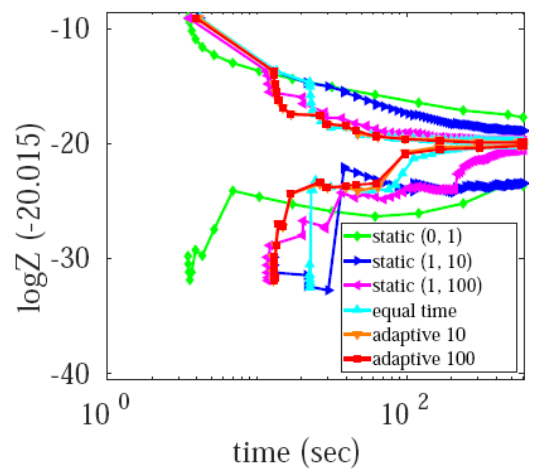
(c) protein/1qsq



(d) protein/1whi

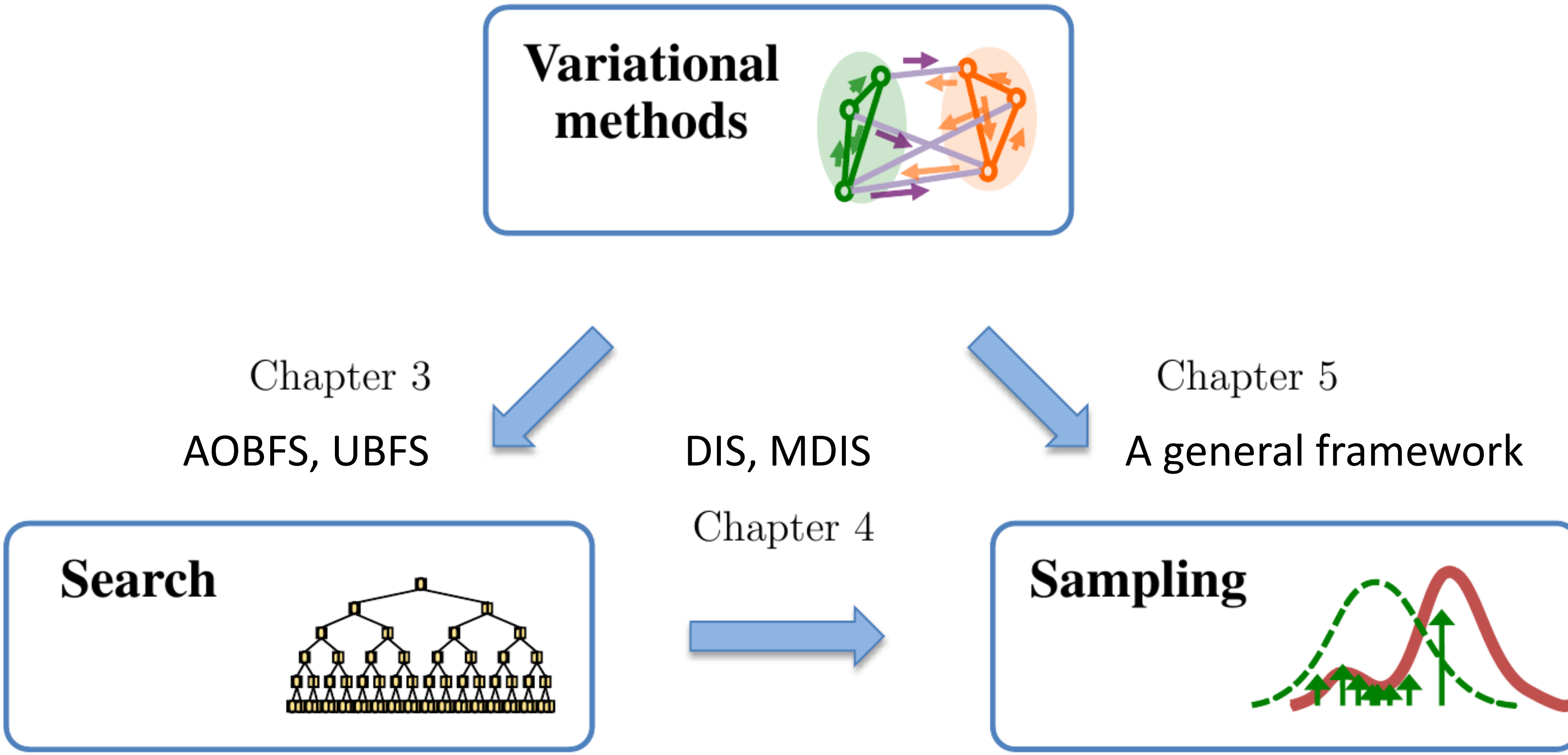


(e) promedas/or_chain_132.fg



(f) promedas/or_chain_153.fg

Conclusions



Future Directions