Partition Hybrid Kernels for MCMC Exploration of Constrained State Spaces

Keith Bonawitz
Research Qualifying Exam – Practice Talk
May 15, 2006
Outline

• **Motivation:** Probabilistic Models in AI
• **Primer:** MCMC
• **Partition Hybrid Kernels**
  – Single Kernels
  – Hierarchy Compilation
  – Empirical Results
Motivation: Probabilistic Models

- Managing Uncertainty
- Natural integration of top-down and bottom-up processing
  - Genesis: Streams and Counterstreams
- Bayes: Generative model → inference model
- Well-defined Composition Semantics
Motivation: Probabilistic Models

The block is behind the ball

"The block is the ball"
Motivation: Probabilistic Models

World model

Vision

Semantics to Express

Words to Utter

What is Heard

The block is behind the ball

"The block is the ball"
Motivation: Probabilistic Models

World Knowledge

- Anticipated # objects, types of objects, etc
- Particular configuration of objects in 3-space
- “Clean” rendering
- Noise-corrupted rendering
Primer: MCMC

- Computational Level: Probabilistic Models
- Algorithmic Level: Markov Chain Monte Carlo
Primer: MCMC

• Inference in Probabilistic Models
  – Variable Sets: Query, Uninterested, Evidence
  – Easy to write down, evaluate: $p(Q, U, E)$

• Bayes Rule: $p(Q, U | E) = \frac{p(Q, U, E)}{p(E)}$

• Integrate: $p(E) = \int \int p(q, u, E) du \, dq$

• Integrate: $p(Q | E) = \int U p(Q, u | E) du$
**Primer: MCMC**

- Monte Carlo approximation

\[ \langle q_i, u_i \rangle \sim P(Q, U | E) \]

\[ \hat{p}_N(Q | E) = \frac{1}{N} \sum_{i=1}^{N} \delta_{q_i}(Q) \]
Primer: MCMC

- Stochastic Walk through State Space
  \[ Q \times U \]

- Time in \( < q, u > \propto p(q, u \| E) \)
**Primer: MCMC**

1. Let \( s_i = \langle q_i, u_i \rangle \) be the state chain.
2. **Markov Assumption:**
   \[
   p(s_{i+1} | s_1 \ldots s_i) = p(s_{i+1} | s_i)
   \]

\[\ldots \quad s_{i-2} \quad s_{i-1} \quad s_i \quad s_{i+1} \]
**Primer: MCMC**

- Let $s_i = \langle q_i, u_i \rangle$ be the state chain
- **Markov Assumption:**
  \[
  p(s_{i+1} | s_1 \ldots s_i) = p(s_{i+1} | s_i)
  \]
  "Transition Kernel"

\[
\cdots S_{i-2} \longrightarrow S_{i-1} \longrightarrow S_i \longrightarrow S_{i+1}
\]
**Primer: MCMC**

- Let \( s_i = \langle q_i, u_i \rangle \) be the state chain.
- **Markov Assumption:**

\[
K(s_{i+1} \mid s_i)
\]

“Transition Kernel”

... \( s_{i-2} \) \( s_{i-1} \) \( s_i \) \( s_{i+1} \)
 Primer: MCMC

- Let \( s_i = \langle q_i, u_i \rangle \) be the state chain
- Markov Assumption:

\[
K(s_{i+1}|s_i)
\]

“Transition Kernel”

\[ p = pK \]

Stationary Distribution
Primer: MCMC

Metropolis-Hastings Recipe

1. Propose: \( s^* \sim q(s^* | s_i) \)

2. Compute: \( A(s_i, s^*) = \min \left\{ 1, \frac{p(s^*)q(s_i | s^*)}{p(s_i)q(s^* | s_i)} \right\} \)

3. With probability \( A(s_i, s^*) \) accept \( s_{i+1} = s^* \), otherwise \( s_{i+1} = s_i \)
Primer: MCMC

Composing Kernels

• If $K_1$ and $K_2$ both target $p(x)$, then so do:

   **Cycle Hybrid** $K_1 K_2$

   **Mixture Hybrid** $\alpha K_1 + (1 - \alpha) K_2$
Partition Hybrids: Intro

- Spheres & Blocks in 3-space
  - Sphere Birth/Death
  - Block Birth/Death
  - Sphere Relocate
  - Block Relocate
  - Block Reorient
Partition Hybrids: Intro

• Spheres & Blocks in 3-space
  - Sphere Birth/Death
  - Block Birth/Death

- Sphere Relocate
  Exist Spheres?

- Block Relocate
- Block Reorient
  Exist Blocks?
Common Approach: Propose outside the State Space

\[ A(s_i, s^*) = \min \left\{ 1, \frac{p(s^*)q(s_i | s^*)}{p(s_i)q(s^* | s_i)} \right\} \]
Partition Hybrids: Intro

- Common Approach: Propose outside the State Space

$$A(s_i, s^*) = \min \left\{ 1, \frac{p(s^*)q(s_i | s^*)}{p(s_i)q(s^* | s_i)} \right\}$$

0 Always Reject

Well Defined?
Partition Hybrids: Definition

Partition Hybrid Kernel → Spheres?

- true: Sphere Relocate
- false: Identity
**Partition Hybrids: Definition**

\[ \hat{K}(s^*|s) = \begin{cases} 
K_1(s^*|s), & \text{if } s \in \Omega_1; \\
K_2(s^*|s), & \text{if } s \in \Omega_2; 
\end{cases} \]
Flip 3 fair coins; \( x = \) number of heads

\[
p_{target}(x) = Binomial(x; 3, 0.5) = \begin{bmatrix} 0.125 & 0.375 & 0.375 & 0.125 \end{bmatrix}
\]
**Partition Hybrids: Requirements**

\[
K_1 = \begin{bmatrix}
0.265 & 0.249 & 0.229 & 0.257 \\
0.083 & 0.709 & 0.204 & 0.004 \\
0.076 & 0.204 & 0.624 & 0.095 \\
0.257 & 0.011 & 0.285 & 0.447
\end{bmatrix}
\]

\[
P_{target} = P_{target} K_1
\]

\[
K_2 = \begin{bmatrix}
0.666 & 0.041 & 0.099 & 0.194 \\
0.014 & 0.864 & 0.121 & 0.002 \\
0.033 & 0.121 & 0.835 & 0.011 \\
0.194 & 0.004 & 0.032 & 0.770
\end{bmatrix}
\]

\[
P_{target} = P_{target} K_2
\]
Partition Hybrids: Requirements

\[
\begin{bmatrix}
0.265 & 0.249 & 0.229 & 0.257 \\
0.083 & 0.709 & 0.204 & 0.004 \\
0.076 & 0.204 & 0.624 & 0.095 \\
0.257 & 0.011 & 0.285 & 0.447 \\
\end{bmatrix}
\quad \begin{bmatrix}
\Omega_1 \\
\Omega_2 \\
\end{bmatrix}
\begin{bmatrix}
0.666 & 0.041 & 0.099 & 0.194 \\
0.014 & 0.864 & 0.121 & 0.002 \\
0.033 & 0.121 & 0.835 & 0.011 \\
0.194 & 0.004 & 0.032 & 0.770 \\
\end{bmatrix}
\]
Partition Hybrids: Requirements

\[ K_{naive} = \begin{bmatrix}
0.265 & 0.249 & 0.229 & 0.257 \\
0.083 & 0.709 & 0.204 & 0.004 \\
0.076 & 0.204 & 0.624 & 0.095 \\
0.257 & 0.011 & 0.285 & 0.447
\end{bmatrix} \]

\[ \Omega_1 = \begin{bmatrix}
0.666 & 0.041 & 0.099 & 0.194 \\
0.014 & 0.864 & 0.121 & 0.002 \\
0.033 & 0.121 & 0.835 & 0.011 \\
0.194 & 0.004 & 0.032 & 0.770
\end{bmatrix} \]
Partition Hybrids: Requirements

\[
p_{\text{target}} \times K_{\text{naive}} = \begin{bmatrix}
0.125 & 0.375 & 0.375 & 0.125 \\
0.101 & 0.343 & 0.422 & 0.138
\end{bmatrix} \neq p_{\text{target}}
\]

\[
K_{\text{naive}} = \begin{bmatrix}
0.265 & 0.249 & 0.229 & 0.257 \\
0.083 & 0.709 & 0.204 & 0.004 \\
0.033 & 0.121 & 0.835 & 0.011 \\
0.194 & 0.004 & 0.032 & 0.770
\end{bmatrix}
\]
No Partition-Crossing Transitions

\[ \forall x \in \Omega_1, x^* \in \Omega_2 : K_1(x^*|x) = 0, K_2(x^*|x) = 0 \]

\[ \forall x \in \Omega_2, x^* \in \Omega_1 : K_1(x^*|x) = 0, K_2(x^*|x) = 0 \]
## Partition Hybrids: Properties

<table>
<thead>
<tr>
<th>Kernel</th>
<th>Stationary Distribution</th>
<th>Partition</th>
<th>Truncated to Partition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_1$</td>
<td>$p_1$</td>
<td>$\Omega_1$</td>
<td>$\hat{p}_1$</td>
</tr>
<tr>
<td>$K_2$</td>
<td>$p_2$</td>
<td>$\Omega_2$</td>
<td>$\hat{p}_2$</td>
</tr>
</tbody>
</table>

**Partition Hybrid Stationary Distribution**

$$\alpha \hat{p}_1 + (1 - \alpha) \hat{p}_2 \quad \forall 0 \leq \alpha \leq 1$$
Partition Hybrids: Properties

- Spheres?
  - true
    - Sphere Relocate
  - false
    - Identity
Partition Hybrids: Hierarchies

Spheres?

true
Sphere Relocate

false
Block Birth/Death
Partition Hybrids: Hierarchies

Spheres?

true

Sphere Relocate

false

Sphere Birth/Death

Partition Hybrid Kernels

Keith Bonawitz
Partition Hybrids: Hierarchies

Spheres?

true

false

Sphere Relocate

Blocks?

true

false

Block Relocate

Identity
Partition Hybrids: Hierarchies

Spheres?
- true
  - Sphere Relocate
- false

Blocks?
- true
  - Identity
  - Block Relocate
- false
  - Block Reorient
Partition Hybrids: Hierarchies

Blocks?

- true
  - true
    - Spheres?
      - true
        - Block Relocate
        - Sphere Relocate
        - Event
        - Sphere Birth/Death
        - Block Reorient
        - Sphere Relocate
        - Sphere Birth/Death
        - Block Birth/Death
        - Sphere Relocate
        - Block Relocate
        - Block Birth/Death
        - Block Reorient
        - Block Birth/Death
      - false
        - Spheres?
        - Identity
        - Sphere Birth/Death
        - Block Reorient
        - Sphere Relocate
        - Sphere Birth/Death
        - Block Relocate
        - Block Birth/Death
        - Block Reorient
        - Block Birth/Death
    - false
      - Spheres?
      - Identity
      - Sphere Birth/Death
      - Block Reorient
      - Sphere Relocate
      - Sphere Birth/Death
      - Block Relocate
      - Block Birth/Death
      - Block Reorient
      - Block Birth/Death

- false
  - false
    - Spheres?
      - true
        - Block Relocate
        - Sphere Relocate
        - Event
        - Sphere Birth/Death
        - Block Reorient
        - Sphere Relocate
        - Sphere Birth/Death
        - Block Birth/Death
        - Sphere Relocate
        - Block Relocate
        - Block Birth/Death
        - Block Reorient
        - Block Birth/Death
      - false
        - Spheres?
        - Identity
        - Sphere Birth/Death
        - Block Reorient
        - Sphere Relocate
        - Sphere Birth/Death
        - Block Relocate
        - Block Birth/Death
        - Block Reorient
        - Block Birth/Death
    - true
      - Spheres?
      - Identity
      - Sphere Birth/Death
      - Block Reorient
      - Sphere Relocate
      - Sphere Birth/Death
      - Block Relocate
      - Block Birth/Death
      - Block Reorient
      - Block Birth/Death

Spheres?

- true
  - Spheres?
    - true
      - Block Relocate
      - Sphere Relocate
      - Event
      - Sphere Birth/Death
      - Block Reorient
      - Sphere Relocate
      - Sphere Birth/Death
      - Block Birth/Death
      - Sphere Relocate
      - Block Relocate
      - Block Birth/Death
      - Block Reorient
      - Block Birth/Death
    - false
      - Spheres?
      - Identity
      - Sphere Birth/Death
      - Block Reorient
      - Sphere Relocate
      - Sphere Birth/Death
      - Block Relocate
      - Block Birth/Death
      - Block Reorient
      - Block Birth/Death
- false
  - false
    - Spheres?
    - Identity
    - Sphere Birth/Death
    - Block Reorient
    - Sphere Relocate
    - Sphere Birth/Death
    - Block Relocate
    - Block Birth/Death
    - Block Reorient
    - Block Birth/Death
Partition Hybrids: Hierarchies

**PH-HIERARCHY** (*Inherited-Partition*)

1. $mixture \leftarrow$ an empty mixture hybrid kernel
2. **for** each base kernel $K$
3.  **do if** COMPATIBLE-BASE-KERNEL?($K$, *Inherited-Partition*)
4.    **then** $\triangleright$ If $K$ is definitely applicable, there is no need to recurse
5.    **if** ($\text{Background-Knowledge} \land \text{Inherited-Partition} \Rightarrow \text{applicable}_K$)
6.       **then** add $K$ to $mixture$
7.    **else** $\text{This-Partition} \leftarrow \text{Inherited-Partition} \land \neg \text{applicable}_K$
8.       $\text{false-subkernel} \leftarrow \text{PH-HIERARCHY}($This-Partition$)$
9. $\triangleright$ Simplify degenerate mixtures
10. **if** COUNT($mixture\.subkernels$) = 0
11.    **then return** the identity kernel
12. **elseif** COUNT($mixture\.subkernels$) = 1
13.    **then return** $mixture$’s only subkernel
14. **else return** $mixture$
COMPATIBLE-BASE-KERNEL? \((K, Inherited-Partition)\)

1. Plausible: Don’t use K if it cannot be applicable in Inherited-Partition
2. if \((Background-Knowledge \land Inherited-Partition \Rightarrow \neg \text{applicable}_K)\)
3. then return FALSE
4. Safe: K is not compatible if it can produce partition-crossing transitions
5. for each \(postcondition \in \text{postconditions}_K\)
6. do \(\text{Strips-Vars} \leftarrow \text{VARS}(Background-Knowledge \cup \text{VARS}(Inherited-Partition) \cup \text{VARS}(\text{applicable}_K))\)
7. if \(\text{Inherited-Partition}\vert_{\text{before}} \land \text{applicable}_K\vert_{\text{before}} \land Background-Knowledge\vert_{\text{before}} \land \text{STRIPS(}\text{Strips-Vars, postcondition}) \land Background-Knowledge\vert_{\text{after}} \Rightarrow (\text{Inherited-Partition}\vert_{\text{after}} \land \text{applicable}_K\vert_{\text{after}})\)
8. then return FALSE
9. return TRUE
Partition Hybrids: Hierarchies

Ergodicity

Partition Hybrid Kernels

Keith Bonawitz
Partition Hybrids: Hierarchies

Ergodicity

Blocks?

Blocks?

Spheres?

false

false

false

true

true

true

Block Relocate

Block Reorient

Block Birth/Death

Sphere Birth/Death

Sphere Relocate

Identity

Identity

Identity
$n_{type} \sim \begin{cases} 
\text{Poisson}(10) & \text{if } type \in \{\text{Thing, Place}\} \\
\text{Poisson}(2) & \text{if } type \in \{\text{PathElem, Path, Event, Cause}\}
\end{cases}$

<table>
<thead>
<tr>
<th>Node Type</th>
<th>Children</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thing</td>
<td>(none)</td>
</tr>
<tr>
<td>Place</td>
<td>$\rightarrow$ Thing</td>
</tr>
<tr>
<td>PathElem</td>
<td>$\rightarrow$ Place</td>
</tr>
<tr>
<td>Path</td>
<td>$\rightarrow$ PathElem $\sim$ \text{Poisson}_{[1,\infty]}(2)$</td>
</tr>
<tr>
<td>Event</td>
<td>$\rightarrow$ Thing, $\rightarrow$ Path</td>
</tr>
<tr>
<td>Cause</td>
<td>$\rightarrow$ Thing, $\rightarrow$ Event</td>
</tr>
</tbody>
</table>
Partition Hybrids: Empirical

Partition Hybrid Kernels

Keith Bonawitz
Partition Hybrids: Empirical

MCMC Approximation: # of Thing Nodes

- MCMC
- Exact
Partition Hybrids: Empirical

**MCMC Approximation: # of Place Nodes**

- **Red** line: MCMC
- **Blue** line: Exact

The graph shows the comparison between MCMC approximation and the exact method for the number of place nodes. The x-axis represents the number of places, and the y-axis represents the probability of having that many place nodes. The peak in the graph indicates the most likely number of place nodes, with MCMC approximation closely matching the exact method.
Partition Hybrids: Empirical

MCMC Approximation: # of PathElem Nodes

- MCMC
- Exact

# of PathElems
Partition Hybrids: Empirical

MCMC Approximation: # of Path Nodes

# of Paths

p(# of Paths)

0 0.05 0.1 0.15 0.2 0.25 0.3 0.35

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

MCMC
Exact
Partition Hybrids: Empirical

MCMC Approximation: # of Event Nodes

![Graph showing the comparison between MCMC and Exact methods for the number of event nodes. The graph plots the probability of having a certain number of event nodes against the number of events. The MCMC method is represented by squares, and the Exact method is represented by triangles. The x-axis represents the number of events, ranging from 0 to 25, and the y-axis represents the probability of having that many event nodes, ranging from 0 to 0.4. The graph shows that the MCMC method approximates the Exact method closely, with both methods showing a peak at 0 events and a rapid decrease as the number of events increases.]
Partition Hybrids: Empirical

Divergence of MCMC-approximated marginalizations and true marginalizations, single run

Number of Samples (* 100)

Jensen-Shannon Divergence (bits)

- # Things
- # Places
- # PathElems
- # Paths
- # Events
- # Causes
Partition Hybrids: Empirical

Divergence of MCMC-approximated marginalizations and true marginalizations, 10 run average

- # Things
- # Places
- # PathElements
- # Paths
- # Events
- # Causes

Number of Samples (* 100)
Improved Mixing

\[ p(K \text{ used at MCMC step } i | \text{hierarchy}) = \int_{\Omega} p_{eq}(x) \cdot p(K | x, \text{hierarchy}) dx \]

- \[ p(Identity | \text{hierarchy}_{id}) = 0.00478 \]
- \[ p(Identity | \text{hierarchy}_{PH}) = 3.84 \cdot 10^{-13} \]

\[ 1.24 \cdot 10^{10} \text{ Improvement factor} \]
Improved Mixing

\( n_{type} \sim \text{Poisson}(0.5) \quad \text{for type} \in \{\text{Thing, Place, PathElem, Path, Event, Cause}\} \)

\[
p(Identity|\text{hierarchy}_{id}) = 0.590
\]

\[
p(Identity|\text{hierarchy}_{PH}) = 0.136
\]

\[4.34 \quad \text{Improvement factor}\]

\[\sim 45,000 \quad \text{In a 100,000 sample run}\]
Contributions

• Characterized challenge of MCMC in a constrained state space
• Proposed novel partition hybrid kernels
• Created algorithm for partition hybrid hierarchies
• Established ergodicity conditions
• Demonstrated better mixing
• Collected empirical evidence of correctness
Motivation: Probabilistic Models

I. Degrees of Plausibility are represented by real numbers

II. Qualitative Correspondence with common sense

\[(A|C') > (A|C) \land (B|AC') = (B|AC) \implies (AB|C') \geq (AB|C) \land (\bar{A}|C') < (\bar{A}|C)\]

III. Consistency

a) Every way of reasoning produces the same result

b) Considers all evidence

c) Equivalent state have equivalent plausibility

\textbf{STRIPS}(\textit{variables, postcondition})

1. \textit{sentence} $\leftarrow$ \textit{postcondition}|\textit{after}
2. \textbf{for} each \textit{var} $\in$ \textit{variables} $-$ \textit{VARS}(\textit{postcondition})
3. \textbf{do} \textit{sentence} $\leftarrow$ \textit{sentence} $\land$ (\textit{var}|\textit{before} $\iff$ \textit{var}|\textit{after})
4. \textbf{return} \textit{sentence}
\[ p_{target}(G) \propto \text{Poisson}(n_{Thing}; 10) \cdot n_{Thing}! \cdot \text{Poisson}(n_{Place}; 10) \cdot n_{Place}! \left( \frac{1}{n_{Thing}} \right)^{n_{Place}} \cdot \text{Poisson}(n_{PathElem}; 2) \cdot n_{PathElem}! \cdot \left( \frac{1}{n_{Place}} \right)^{n_{PathElem}} \cdot \text{Poisson}(n_{Path}; 2) \cdot n_{Path}! \cdot \prod_{i}^{n_{Path}} \text{Poisson}_{[1, \infty)}(n_{Path_i}; 2) \left( \frac{1}{n_{PathElem}} \right)^{n_{Path_i}} \cdot \text{Poisson}(n_{Event}; 2) \cdot n_{Event}! \cdot \left( \frac{1}{n_{Thing}} \right)^{n_{Event}} \cdot \left( \frac{1}{n_{Path}} \right)^{n_{Event}} \cdot \text{Poisson}(n_{Cause}; 2) \cdot n_{Cause}! \cdot \left( \frac{1}{n_{Thing}} \right)^{n_{Cause}} \cdot \left( \frac{1}{n_{Event}} \right)^{n_{Cause}} \]
## Partition Hybrids: Empirical

<table>
<thead>
<tr>
<th>Kernel Node Type</th>
<th>Applicability Condition</th>
<th>Postcondition Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thing</td>
<td>TRUE</td>
<td>{existThing, \neg existThing}</td>
</tr>
<tr>
<td>Place</td>
<td>existThing</td>
<td>{existPlace, \neg existPlace}</td>
</tr>
<tr>
<td>PathElem</td>
<td>existPlace</td>
<td>{existPathElem, \neg existPathElem}</td>
</tr>
<tr>
<td>Path</td>
<td>existPathElem</td>
<td>{existPath, \neg existPath}</td>
</tr>
<tr>
<td>Event</td>
<td>existThing \land existPath</td>
<td>{existEvent, \neg existEvent}</td>
</tr>
<tr>
<td>Cause</td>
<td>existThing \land existEvent</td>
<td>{existCause, \neg existCause}</td>
</tr>
</tbody>
</table>

**background** = (existPlace \Rightarrow existThing) \land (existPathElem \Rightarrow existPlace) 
\land (existPath \Rightarrow existPathElem) \land (existEvent \Rightarrow existThing) 
\land (existEvent \Rightarrow existPath) \land (existCause \Rightarrow existThing) 
\land (existCause \Rightarrow existEvent)
Partition Hybrids: Empirical

\[ D_{KL}(p|q) = \int_{\text{supp}\, p} p(x) \log \frac{p(x)}{q(x)} \, dx \]

\[ D_{JS}(p, q) = \frac{1}{2} (D_{KL}(p|\frac{p+q}{2}) + D_{KL}(q|\frac{p+q}{2})) \]
MCMC Approximation: # of Thing Nodes

\[ n_{\text{type}} \sim \text{Poisson}(0.5) \quad \text{for type} \in \{\text{Thing, Place, PathElem, Path, Event, Cause}\} \]
Partition Hybrids: Empirical

MCMC Approximation: # of Place Nodes

\[ n_{\text{type}} \sim \text{Poisson}(0.5) \text{ for type } \in \{\text{Thing}, \text{Place}, \text{PathElem}, \text{Path}, \text{Event}, \text{Cause}\} \]
MCMC Approximation: # of PathElem Nodes

\[ n_{\text{type}} \sim \text{Poisson}(0.5) \quad \text{for} \quad \text{type} \in \{ \text{Thing, Place, PathElem, Path, Event, Cause} \} \]
Partition Hybrids: Empirical

MCMC Approximation: # of Path Nodes

\[ n_{\text{type}} \sim \text{Poisson}(0.5) \quad \text{for type} \in \{ \text{Thing, Place, PathElem, Path, Event, Cause} \} \]
$n_{type} \sim \text{Poisson}(0.5)$ for $\text{type} \in \{\text{Thing, Place, PathElem, Path, Event, Cause}\}$
Partition Hybrids: Empirical

MCMC Approximation: # of Cause Nodes

\[ n_{\text{type}} \sim \text{Poisson}(0.5) \quad \text{for type} \in \{\text{Thing, Place, PathElem, Path, Event, Cause}\} \]
Partition Hybrids: Empirical

Divergence of MCMC-approximated marginalizations and true marginalizations, single run

\[ n_{\text{type}} \sim \text{Poisson}(0.5) \quad \text{for type} \in \{\text{Thing, Place, PathElem, Path, Event, Cause}\} \]
Partition Hybrids: Empirical

Divergence of MCMC-approximated marginalizations and true marginalizations, 10 run average

\[ n_{\text{type}} \sim \text{Poisson}(0.5) \quad \text{for type} \in \{\text{Thing, Place, PathElem, Path, Event, Cause}\} \]