**Efficient Monte Carlo sampler for detecting parametric objects in large scenes**

Yannick Verdie and Florent Lafarge

INRIA, Sophia Antipolis

---

**Problem statement**

**Goal**

Detecting parametric objects in large scenes.

---

**Point process**

A point process describes random configurations of points in a continuous bounded set \( \mathbb{X} \).

The most popular family of point processes corresponds to the Markov point processes of objects specified by Gibbs energies on \( \mathbb{C} \) of the form

\[
\forall \in \mathbb{C}, \quad U(x) = \sum_{x \in \mathbb{C}} D(x) + \sum_{x \in x_j} V(x, x_j)
\]

---

**Simulation**

Existing samplers, relying on sequential schemes, suffer from average performances in terms of computation time and stability.

---

<table>
<thead>
<tr>
<th>Standard MCMC sampler</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Initialize ( X_0 = x_0 ) and ( T_0 ) at ( t = 0 );</td>
</tr>
<tr>
<td>2. At iteration ( t ), with ( X_t = x_t ),</td>
</tr>
<tr>
<td>- Choose a subkernel ( Q_{x,t} ) according to probability ( p_{x,t} );</td>
</tr>
<tr>
<td>- Perturb ( x ) to ( y ) according to ( Q_{x,t} (x \rightarrow y) );</td>
</tr>
<tr>
<td>- Compute the Green ratio</td>
</tr>
<tr>
<td>( R = \frac{Q_{x,t}(y \rightarrow x)}{Q_{x,t}(x \rightarrow y)} \exp \left( \frac{U(y) - U(x)}{\beta} \right) )</td>
</tr>
<tr>
<td>- Choose ( X_{t+1} = y ) with probability ( \min(1, R) ), and ( X_{t+1} = x ) otherwise</td>
</tr>
</tbody>
</table>

---

**Our Parallel Markov Chain Monte Carlo algorithm**

**Sampling in parallel**

Thanks to the Markovian properties of the energy \( U(x) \), simultaneous perturbations can be performed at locations far enough apart to not interfere and break the convergence properties.

The cells can be regrouped into \( 2^n \) sets such that each cell is adjacent to cells belonging to different sets of a mic-set.

**Data-driven space-partitioning tree**

We propose a 1-to-\( 2^n \) hierarchical subdivision scheme by building a partitioning tree. The subdivision of the cells is driven by the data.

A quadtree is created so that the levels are recursively partitioned according to the class of interest. The partitioning tree allows the creation of a non uniform point distribution naturally and efficiently.

**Sampler formulation**

**Our parallel sampler**

1. Initialize \( X_0 = x_0 \) and \( T_0 \) at \( t = 0 \);
2. Compute a space-partitioning tree \( T \);
3. At iteration \( t \), with \( X_t = x_t \),
   - Choose a mic-set \( S_{x,t} \) \( \in \mathbb{X} \) and a kernel type \( \mathcal{T} \) according to probability \( \mathcal{Q}_{c,t}(x \rightarrow \mathcal{Q}) \);
   - For each cell \( c \in S_{x,t} \),
     - Perturb \( x \) in the cell \( c \) to a configuration according to \( Q_{c,t}(x \rightarrow y) \);
   - Calculate the Green ratio
     \[
     R = \frac{Q_{c,t}(y \rightarrow x)}{Q_{c,t}(x \rightarrow y)} \exp \left( \frac{U(y) - U(x)}{\beta} \right)
     \]
   - Choose \( X_{t+1} = y \) with probability \( \min(1, R) \), and \( X_{t+1} = x \) otherwise

---

**Experiments**

**Line-networks extraction**

- **Population counting**
- **3D object extraction**
- **Markov Random Field**

---

**Table**

- **Efficient Monte Carlo sampler**
- **RJMCMC**
- **Multiple birth and death**
- **Our sampler without partitioning tree**
- **Our sampler with partitioning tree**

<table>
<thead>
<tr>
<th>Method</th>
<th>Time (%)</th>
<th>#段 (%)</th>
<th>#物体 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard MCMC</td>
<td>7.3%</td>
<td>4.2%</td>
<td>1.7%</td>
</tr>
<tr>
<td>RJMCMC</td>
<td>6.35%</td>
<td>6.2%</td>
<td>1.6%</td>
</tr>
<tr>
<td>Our sampler without</td>
<td>7.4%</td>
<td>6.2%</td>
<td>1.6%</td>
</tr>
<tr>
<td>Our sampler with</td>
<td>4.4%</td>
<td>1.8%</td>
<td>1.1%</td>
</tr>
</tbody>
</table>

---

**Figure**

- **2D and 3D images**
- **3D scans**
- **3D trees**
- **Point processes**
- **Markov Point processes**
- **Markov Point process of line-segments**

---

**Algorithm**

1. Initialize \( X_0 = x_0 \) and \( T_0 \) at \( t = 0 \);
2. Compute a space-partitioning tree \( T \);
3. At iteration \( t \), with \( X_t = x_t \),
   - Choose a mic-set \( S_{x,t} \) \( \in \mathbb{X} \) and a kernel type \( \mathcal{T} \) according to probability \( \mathcal{Q}_{c,t}(x \rightarrow \mathcal{Q}) \);
   - For each cell \( c \in S_{x,t} \),
     - Perturb \( x \) in the cell \( c \) to a configuration according to \( Q_{c,t}(x \rightarrow y) \);
   - Calculate the Green ratio
     \[
     R = \frac{Q_{c,t}(y \rightarrow x)}{Q_{c,t}(x \rightarrow y)} \exp \left( \frac{U(y) - U(x)}{\beta} \right)
     \]
   - Choose \( X_{t+1} = y \) with probability \( \min(1, R) \), and \( X_{t+1} = x \) otherwise