The **BisimDist** Library

*Efficient Computation of Bisimilarity Distances for Markovian Models*

**Giorgio Bacci**, Giovanni Bacci, Kim G. Larsen, Radu Mardare

Dept. of Computer Science, Aalborg University

QEST 2013
27-30 August, Buenos Aires - Argentina
Motivations

\[
\begin{align*}
S_1 \quad & \frac{2}{3} - \epsilon \\
\quad & \frac{1}{3} + \epsilon \\
S_3 \quad & 1 \\
S_4 \quad & \frac{1}{3} \\
S_5 \quad & \frac{1}{3} \\
S_2 \quad & \frac{1}{3}
\end{align*}
\]
Motivations
Pseudometrics $d : S \times S \rightarrow \mathbb{R}_{\geq 0}$ are the quantitative analogue of an equivalence relation.

<table>
<thead>
<tr>
<th>equivalence</th>
<th>pseudometric</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s \equiv s$</td>
<td>$d(s, s) = 0$</td>
</tr>
<tr>
<td>$s \equiv t \implies t \equiv s$</td>
<td>$d(s, t) = d(t, s)$</td>
</tr>
<tr>
<td>$s \preceq u \land u \preceq t \implies s \preceq t$</td>
<td>$d(s, u) + d(u, t) \geq d(s, t)$</td>
</tr>
</tbody>
</table>
Pseudometrics $d : S \times S \rightarrow \mathbb{R}_{\geq 0}$ are the quantitative analogue of an equivalence relation.

<table>
<thead>
<tr>
<th>equivalence</th>
<th>pseudometric</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s \equiv s$</td>
<td>$d(s, s) = 0$</td>
</tr>
<tr>
<td>$s \equiv t \Rightarrow t \equiv s$</td>
<td>$d(s, t) = d(t, s)$</td>
</tr>
<tr>
<td>$s \cong u \land u \cong t \Rightarrow s \cong t$</td>
<td>$d(s, u) + d(u, t) \geq d(s, t)$</td>
</tr>
</tbody>
</table>

**Bisimilarity Pseudometrics**

$$d(s, t) = 0 \iff s \sim t$$
Pseudometrics on Markovian Models

Markov Chains:
+ pseudometrics of Desharnais et al. [TCS’04]
+ fixed point def. by van Breugel and Worrell [LMCS’08]

Remarkable properties

\[
\sup_{\varphi \in \text{LTL}} | Pr(s \models \varphi) - Pr(t \models \varphi) | \leq d_{\text{MC}}(s, t)
\]

Markov Decision Processes:
+ pseudometrics of Ferns et al. [UAI’04] (fixed point def.)

Remarkable properties

\[
| V^*(s) - V^*(t) | \leq d_{\text{MDP}}(s, t)
\]
Pseudometrics on Markovian Models

**Markov Chains:**
+ pseudometrics of Desharnais et al. [TCS’04]
+ fixed point def. by van Breugel and Worrell [LMCS’08]

Remarkable properties

\[ \sup_{\varphi \in \text{LTL}} |Pr(s \models \varphi) - Pr(t \models \varphi)| \leq d_{MC}(s, t) \]

**Markov Decision Processes:**
+ pseudometrics of Ferns et al. [UAI’04] (fixed point def.)

Remarkable properties

\[ |V^*(s) - V^*(t)| \leq d_{MDP}(s, t) \]
Applications of the pseudometrics

**Model Reduction:** clustering states which are close enough

**Abstraction Testing:** analytical testing of model abstractions

**Parameters Estimation:** baricentrum as the optimal

**Model Prediction:** closest to the ‘optimal’ (usually, not sound)

Bisimilarity pseudometrics have been extensively used in AI

- Policy transfer — Castro, Precup [AAAI’10]
- Basis function discovery — Comanici, Precup [AAAI’11]
- Automatic inference of temporally extended actions
  — Castro, Precup [RL’11]
Applications of the pseudometrics

**Model Reduction:** clustering states which are close enough

**Abstraction Testing:** analytical testing of model abstractions

**Parameters Estimation:** baricentrum as the optimal

**Model Prediction:** closest to the ‘optimal’ (usually, not sound)

Bisimilarity pseudometrics have been extensively used in AI

+ Policy transfer — Castro, Precup [AAAI’10]
+ Basis function discovery — Comanici, Precup [AAAI’11]
+ Automatic inference of temporally extended actions — Castro, Precup [RL’11]
Applications of the pseudometrics

**Model Reduction:** clustering states which are close enough

**Abstraction Testing:** analytical testing of model abstractions

**Parameters Estimation:** baricentrum as the optimal

**Model Prediction:** closest to the ‘optimal’ (usually, not sound)

Bisimilarity pseudometrics have been extensively used in AI

- Policy transfer — Castro, Precup [AAAI’10]
- Basis function discovery — Comanici, Precup [AAAI’11]
- Automatic inference of temporally extended actions — Castro, Precup [RL’11]
Applications of the pseudometrics

**Model Reduction:** clustering states which are close enough

**Abstraction Testing:** analytical testing of model abstractions

**Parameters Estimation:** baricentrum as the optimal

**Model Prediction:** closest to the ‘optimal’ (usually, not sound)

Bisimilarity pseudometrics have been extensively used in AI

- Policy transfer — Castro, Precup [AAAI’10]
- Basis function discovery — Comanici, Precup [AAAI’11]
- Automatic inference of temporally extended actions — Castro, Precup [RL’11]
Applications of the pseudometrics

**Model Reduction:** clustering states which are close enough

**Abstraction Testing:** analytical testing of model abstractions

**Parameters Estimation:** baricentrum as the optimal

**Model Prediction:** closest to the ‘optimal’ (usually, not sound)

Bisimilarity pseudometrics have been extensively used in AI

- Policy transfer — Castro, Precup [AAAI’10]
- Basis function discovery — Comanici, Precup [AAAI’11]
- Automatic inference of temporally extended actions
  — Castro, Precup [RL’11]
Applications of the pseudometrics

**Model Reduction:** clustering states which are close enough

**Abstraction Testing:** analytical testing of model abstractions

**Parameters Estimation:** baricentrum as the optimal

**Model Prediction:** closest to the ‘optimal’ (usually, not sound)

---

**Bisimilarity pseudometrics have been extensively used in AI**

+ Policy transfer — Castro, Precup [AAAI’10]
+ Basis function discovery — Comanici, Precup [AAAI’11]
+ Automatic inference of temporally extended actions — Castro, Precup [RL’11]
Existing methods for computing the distance

Iterative Methods (approximated)

+ based on a fixed point characterization of the pseudometric
+ **Markov Chains** — van Breugel, Worrell [LMCS’08]
+ **Markov Decision Processes** — Ferns et al. [UAI’04]

Iterative + Heuristics (approximated)

+ focus on states where the impact is expected to be greater
+ (similar to asynchronous dynamic programming)

Linear Programming (exact)

+ solution of a linear program with exponentially many constraints
+ ellipsoid method $\Rightarrow$ polynomial
Existing methods for computing the distance

**Iterative Methods**

+ based on a fixed point characterization of the pseudometric
+ **Markov Chains** — van Breugel, Worrell [LMCS’08]
+ **Markov Decision Processes** — Ferns et al. [UAI’04]

**Iterative + Heuristics** — Comanici et al. [QEST’12] (approximated)

+ focus on states where the impact is expected to be greater
+ (similar to asynchronous dynamic programming)

**Linear Programming** — Chen et al. [FoSSaCS’12] (exact)

+ solution of a linear program with exponentially many constraints
+ ellipsoid method  $\Rightarrow$ polynomial
Existing methods for computing the distance

**Iterative Methods** (approximated)
- based on a fixed point characterization of the pseudometric
- **Markov Chains** — van Breugel, Worrell [LMCS’08]
- **Markov Decision Processes** — Ferns et al. [UAI’04]

**Iterative + Heuristics** (approximated)
- focus on states where the impact is expected to be greater
- (similar to asynchronous dynamic programming)

**Linear Programming** (exact)
- solution of a linear program with exponentially many constraints
- ellipsoid method $\Rightarrow$ polynomial
On-the-fly Algorithms

All existing methods require to explore the entire state space.

What if we only need $d(s, t)$?
(can we avoid computation of $d$ on all pairs of states?)
All existing methods require to explore the entire state space.

What if we only need \( d(s, t) \)?
(can we avoid computation of \( d \) on all pairs of states?)

we proposed an on-the-fly algorithm:
+ lazy exploration of \( M \) (only where and when is needed)
+ save computational cost (time & space)
All existing methods require to explore the entire state space.

What if we only need $d(s, t)$?
(can we avoid computation of $d$ on all pairs of states?)

we proposed an **on-the-fly** algorithm:
+ lazy exploration of $M$ (only where and when is needed)
+ save computational cost (time & space)

Markov Chains
[TACAS’13]

Markov Decision Processes
[MFCS’13]
<table>
<thead>
<tr>
<th># States</th>
<th>On-the-Fly (exact)</th>
<th>Iterating (approximated)</th>
<th>Approx. Error*</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time (s)</td>
<td># TPs</td>
<td>Time (s)</td>
</tr>
<tr>
<td>5</td>
<td>0.019</td>
<td>1.191</td>
<td>0.0389</td>
</tr>
<tr>
<td>6</td>
<td>0.059</td>
<td>3.046</td>
<td>0.092</td>
</tr>
<tr>
<td>7</td>
<td>0.138</td>
<td>6.011</td>
<td>0.204</td>
</tr>
<tr>
<td>8</td>
<td>0.255</td>
<td>8.561</td>
<td>0.364</td>
</tr>
<tr>
<td>9</td>
<td>0.499</td>
<td>12.042</td>
<td>0.673</td>
</tr>
<tr>
<td>10</td>
<td>1.003</td>
<td>18.733</td>
<td>1.272</td>
</tr>
<tr>
<td>11</td>
<td>2.159</td>
<td>25.973</td>
<td>2.661</td>
</tr>
<tr>
<td>12</td>
<td>4.642</td>
<td>34.797</td>
<td>5.522</td>
</tr>
<tr>
<td>13</td>
<td>6.735</td>
<td>39.958</td>
<td>8.061</td>
</tr>
<tr>
<td>14</td>
<td>6.336</td>
<td>38.005</td>
<td>7.188</td>
</tr>
<tr>
<td>17</td>
<td>11.261</td>
<td>47.014</td>
<td>12.805</td>
</tr>
<tr>
<td>19</td>
<td>26.635</td>
<td>61.171</td>
<td>29.654</td>
</tr>
<tr>
<td>20</td>
<td>34.379</td>
<td>66.457</td>
<td>38.206</td>
</tr>
</tbody>
</table>

\( (*) \epsilon = \max_{s, t \in S} \delta_\lambda (s, t) - d(s, t) \)
## Empirical Results

### (single-pair)

<table>
<thead>
<tr>
<th># States</th>
<th>out-degree = 3</th>
<th>2 ≤ out-degree ≤ # States</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time (s)</td>
<td># TPs</td>
</tr>
<tr>
<td>5</td>
<td>0.006</td>
<td>0.273</td>
</tr>
<tr>
<td>6</td>
<td>0.012</td>
<td>0.549</td>
</tr>
<tr>
<td>7</td>
<td>0.017</td>
<td>0.981</td>
</tr>
<tr>
<td>8</td>
<td>0.025</td>
<td>1.346</td>
</tr>
<tr>
<td>9</td>
<td>0.026</td>
<td>1.291</td>
</tr>
<tr>
<td>10</td>
<td>0.058</td>
<td>2.038</td>
</tr>
<tr>
<td>11</td>
<td>0.077</td>
<td>1.827</td>
</tr>
<tr>
<td>12</td>
<td>0.043</td>
<td>1.620</td>
</tr>
<tr>
<td>13</td>
<td>0.060</td>
<td>1.882</td>
</tr>
<tr>
<td>14</td>
<td>0.089</td>
<td>2.794</td>
</tr>
</tbody>
</table>
BisimDist is a Mathematica® library that provides two packages:

**MCDist**  
+ Data structures (model definition)

**MDPDist**  
+ Data structure manipulators & visualizers
+ Procedure for computing bisimilarity distances *(on-the-fly!)*
  + approximated methods (from known upper-bounds)
  + future-discount
+ bisimilarity classes / quotient by bisimilarity

Library + Tutorials

http://people.cs.aau.dk/giovbacci/tools.html