Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Representation Discovery for MDPs Using Bisimulation Metrics

Sherry Shanshan Ruan, Gheorghe Comanici, Prakash Panangaden, and **Doina Precup**
School of Computer Science
McGill University, Montreal, QC, Canada
{sherry, gcoman, prakash, dprecup}@cs.mcgill.ca

Solving large sequential decision problems modeled as Markov Decision Processes (MDPs) requires representing the state space by approximations such as state aggregation, linear function approximation or kernel-based methods. In this work, we are mainly interested in state aggregation in which the state space is partitioned into disjoint subsets and values are associated with each partition. Our goal is to construct a partition incrementally in such a way as to provide a good approximation to the true value function.

One approach for this problem is to use bisimulation relations (Givan, Dean, and Greig 2003) or their relaxation as *bisimulation metrics* (Ferns, Panangaden, and Precup 2004). Generally speaking, two states are bisimilar if one can simulate all the transitions of the other and the next state distributions are the same over bisimulation classes. Bisimulation metrics are a quantitative analogue of bisimulation relations, in the sense that they assign a non-negative value to all pairs of states to represent distance between them. These metrics are particularly attractive because they allow quantifying the approximation error for *any* state space partitioning, or more generally, any linear function approximator (Comanici and Precup 2012). However, bisimulation metric computation is very expensive. Indeed, (Ferns and Precup 2014) show that computing the metric amounts to solving an MDP resulting from a coupling of the state space with itself, and such a coupling has size quadratic in the number of states.

In this work, we tackle this problem by proposing a significant improvement in the computation of bisimulation metrics. Our first contribution is a construct of an iteratively improving sequence of state space partitions which converges in the limit to the bisimulation relation. We prove that after each iteration, the error of the value function computed over this partition (compared to the true optimal value function) is bounded. We define a partition $\mathcal{B}$ as a basis $\{\phi_i\}_{i=1}^m$ such that $\phi_i \in \{0,1\}$ and $\sum_i \phi_i(s) = 1, \forall$ state $s \in$ state space S. Then the partition procedure is described in Algorithm 1 where $\mathcal{P}^a$ is associated with the policy choosing action a deterministically. The algorithm has complexity

$$O(\textstyle\sum_{\phi \in B_{n-1}}(\phi^T\phi)|B_\phi||A| + |B_n|^2|B_{n-1}|^2 \log |B_{n-1}||A|)$$

where the first term of the sum accounts for the construction of B_n and the second part of the sum accounts for the computation of the metric d_n. As n approaches ∞, the update algorithm runs in $O(|A|(|S||B_\sim| + |B_\sim|^4 \log |B_\sim|))$, which is an upper bound for the update at any step.

Algorithm 1 Partition declustering

```
Given a partition $B_n, n \geq 1$
$B_{n+1} \leftarrow \emptyset$
for all $\phi \in B_n$ do
    $B_\phi \leftarrow \emptyset$
    for all $s$ with $\phi(s) = 1$ do
        for all $\phi' \in B_\phi$ do
            choose $s'$ with $\phi'(s') = 1$
            if $\forall a, \forall \phi'' \in B_{n-1}, (\mathcal{P}^a\phi'')(s) = (\mathcal{P}^a\phi'')(s')$
             then $\phi'(s) \leftarrow 1$
        end for
        if $\phi'(s) = 0,\ \forall \phi' \in B_\phi$
         then add a new element $\hat{\phi}$ to $B_\phi$ and set $\hat{\phi}(s) = 1$
    end for
    add the elements of $B_\phi$ to $B_{n+1}$
end for
```

This approach can provide substantial space and computation time savings since the value function approximation at each step is computed over partitions rather than states. We illustrate the improvement by computing bisimulation metrics over a series of MDPs (the well-known *Puddle World* problem) that increase in size but remain same in structures. The key finding is that after some point, the number of features found is almost constant even when the state space increases, as demonstrated in the top panel of Figure 1. This is because the complexity of the reward function and the transition system remain unchanged. The bottom panel of Figure 1 shows the runtime of computing bisimulation metrics over partitions for up to 14 iterations of Algorithm 1. These empirical results illustrate that the computational complexity of computing bisimulation-based representations and corresponding metrics is mostly dependent on the intrinsic complexity of the reward function and transition models.

The second contribution of our work is an algorithm for asynchronous updates of the metric and the representation. Motivated by asynchronous approach proposed in (Comanici, Panangaden, and Precup 2012), our asynchronous algorithm attempts to maintain the computational cost away from the latter upper bound. Similar to the synchronous one, the asynchronous partition algorithm generates a sequence

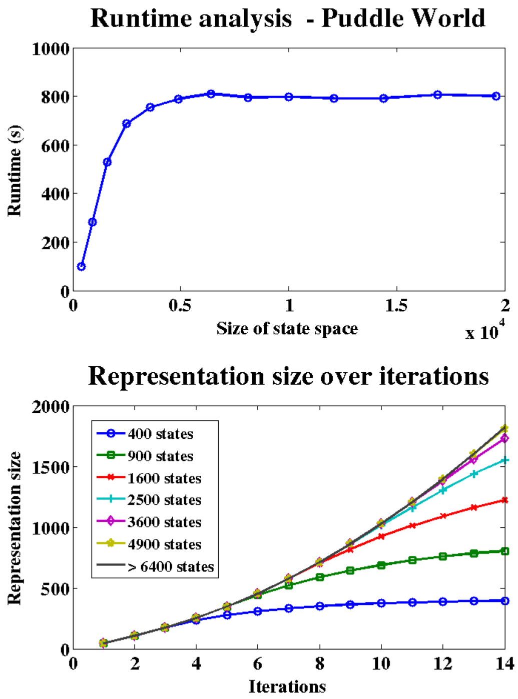

Figure 1: ***Puddle World - computing the metric.*** **Top:** A plot of the runtime as a function of state space size when computing the metric. If metrics are computed over the state space instead, the runtime jumps from 129 seconds on a 400 states environment, to 1375 seconds when the number of states is 1600. **Bottom:** The number of features in the intermediate steps of the algorithm. Note that for state spaces larger than 4900, the number of features does not change substantially with the size of the space.

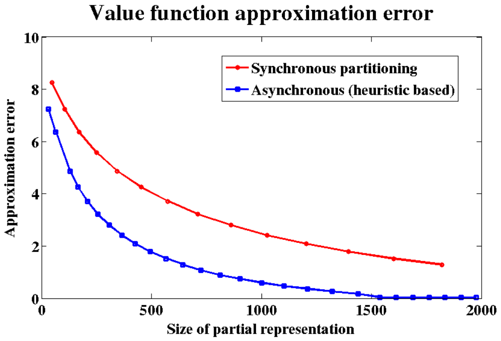

Figure 2: **Asynchronous computation:** A plot of the approximation error in the value function computation (L_∞ norm) as the size of the alternative representation is increased. This particular plot was generated on a Puddle World of size 4900.

of partitions and metric over them with the property that the corresponding metrics can be transformed to a sequence of metrics converging to the desired Kantorovich-based fixed point bisimulation metric. We provide theoretical conditions which allow computational effort to be focused on parts of the state space where changes are happening rapidly, similar to successful asynchronous or distributed dynamic programming techniques such as (Bertsekas and Tsitsiklis 1996).

To assess the importance of the asynchronous partition and metric update, we fixed the size of the Puddle World and compared the value function approximation error as a function of the size of intermediate representations. For each partition we performed dynamic programming to compute the value function and used a heuristic which selects the largest block first to update the partition and metric. This comes from the intuition that it would be desirable to seek a representation that is as uniform as possible. As can be seen in Figure 2, the asynchronous algorithm reaches representations of higher quality at much earlier stages of the iterative framework. This empirical result illustrates the use of heuristics can substantially speed up the computation.

In conclusion, we presented two new ways of describing bisimulation metrics from a theoretical perspective, and we used these to design novel iterative refinement algorithms. These algorithms provide substantial improvement in terms of time and memory usage, and more flexibility in terms of guiding the search for alternative state space representations for MDPs. As illustrated, the methods we propose are barely as sensitive to the size of the state space of the problem.

The approach presented in this paper opens the door to more specialized strategies to finding bisimulation-based MDP representations. We illustrated the advantage of using heuristic based search strategies, but the strategy we used (which attempts to keep the size of state partitions roughly the same) is very simple, and it is likely that more sophisticated approaches would work better. For example, one could try strategies similar to prioritized sweeping, which focus on areas of the state space where the metric is changing drastically. Investigating more sophisticated heuristics and applying them to larger problems is a worthwhile approach for future work.

References

Bertsekas, D. P., and Tsitsiklis, J. N. 1996. *Neuro-Dynamic Programming*. Athena Scientific, Bellman, MA.

Comanici, G., and Precup, D. 2012. Basis function discovery using spectral clustering and bisimulation metrics. In *Lecture Notes in Computer Science, Volume 7113*.

Comanici, G.; Panangaden, P.; and Precup, D. 2012. On-the-fly algorithms for bisimulation metrics. In *the 9th International Conference on Quantitative Evaluation of SysTems (QEST)*.

Ferns, N., and Precup, D. 2014. Bisimulation metrics are optimal value functions. In *the 30th Conference on Uncertainty in Artificial Intelligence*.

Ferns, N.; Panangaden, P.; and Precup, D. 2004. Metrics for finite Markov decision precesses. In *the 20th Conference on Uncertainty in Artificial Intelligence*, 162–169.

Givan, R.; Dean, T.; and Greig, M. 2003. Equivalence notions and model minimization in Markov decision processes. *Artificial Intelligence* 147(1-2):163–223.