Approximating the Permanent by Sampling from Adaptive Partitions

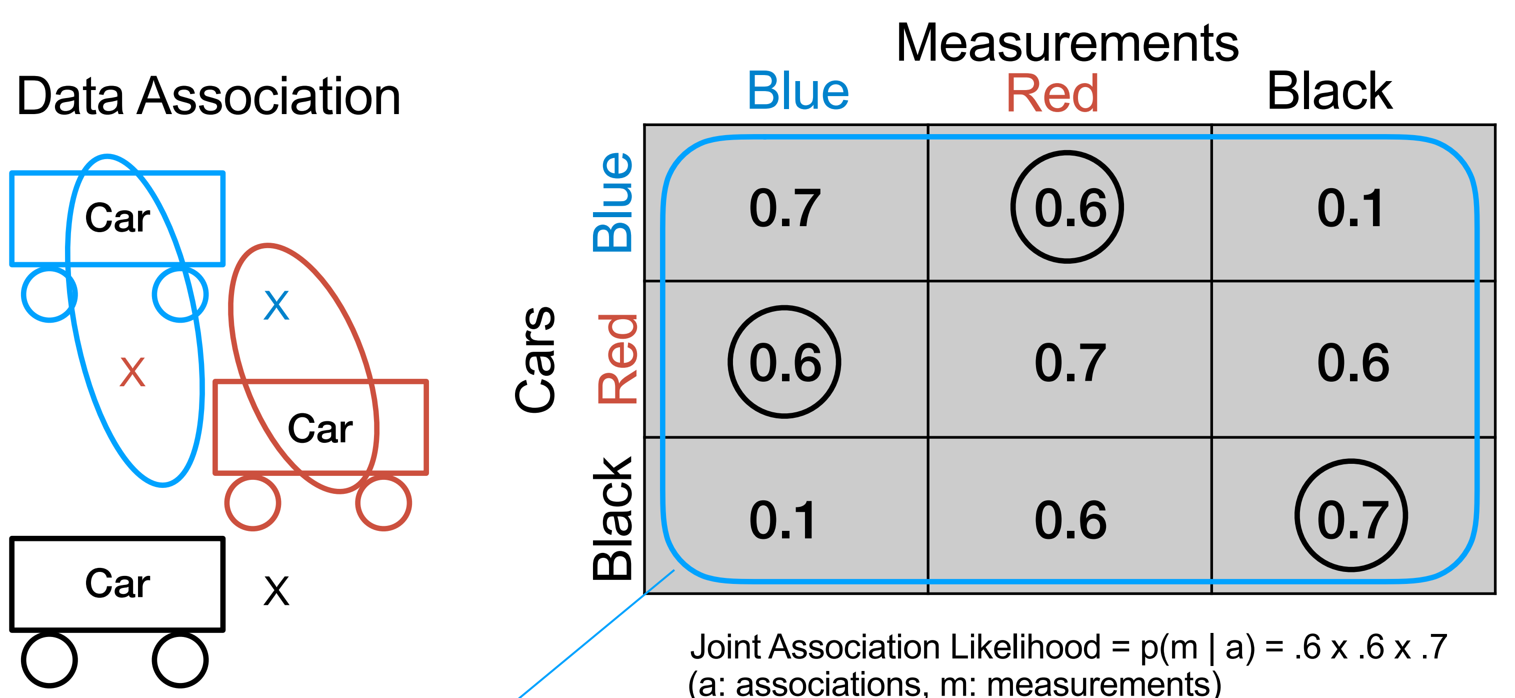
Jonathan Kuck, Tri Dao, Hamid Rezatofighi, Ashish Sabharwal, and Stefano Ermon

kuck@stanford.edu, trid@stanford.edu, hamidrt@stanford.edu, ashishs@allenai.org, ermon@stanford.edu

TLDR: Computing the matrix permanent exactly is intractable. We introduce a sampling based approximation that adaptively partitions the state space of permutations. Our method has a polynomial runtime guarantee on dense matrices, gives speedups of over 10x on real world matrices, and can be used to improve the sample efficiency of multi-target tracking algorithms.

Problem: Estimating the matrix permanent.
Algorithm Outputs: exact samples of permutations from the distribution defined by the matrix permanent along with high probability bounds on the permanent.

Motivation: Multi-Object Target Tracking



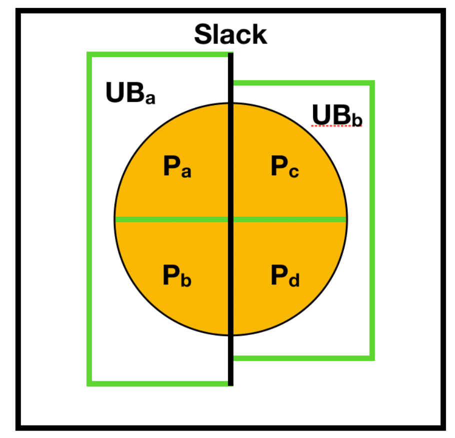
Joint Association Likelihood = p(m | a) = .6 x .6 x .7
(a: associations, m: measurements)
Posterior distribution over joint associations:

Association Matrix *Permanent*

$$p(\mathrm{a}|\mathrm{m}) = \frac{p(\mathrm{m}|\mathrm{a})p(\mathrm{a})}{p(\mathrm{m})} \qquad p(\mathrm{m}) = \sum_{\sigma \in S_n} \prod_{j=1}^{n} A(j, \sigma(j))$$

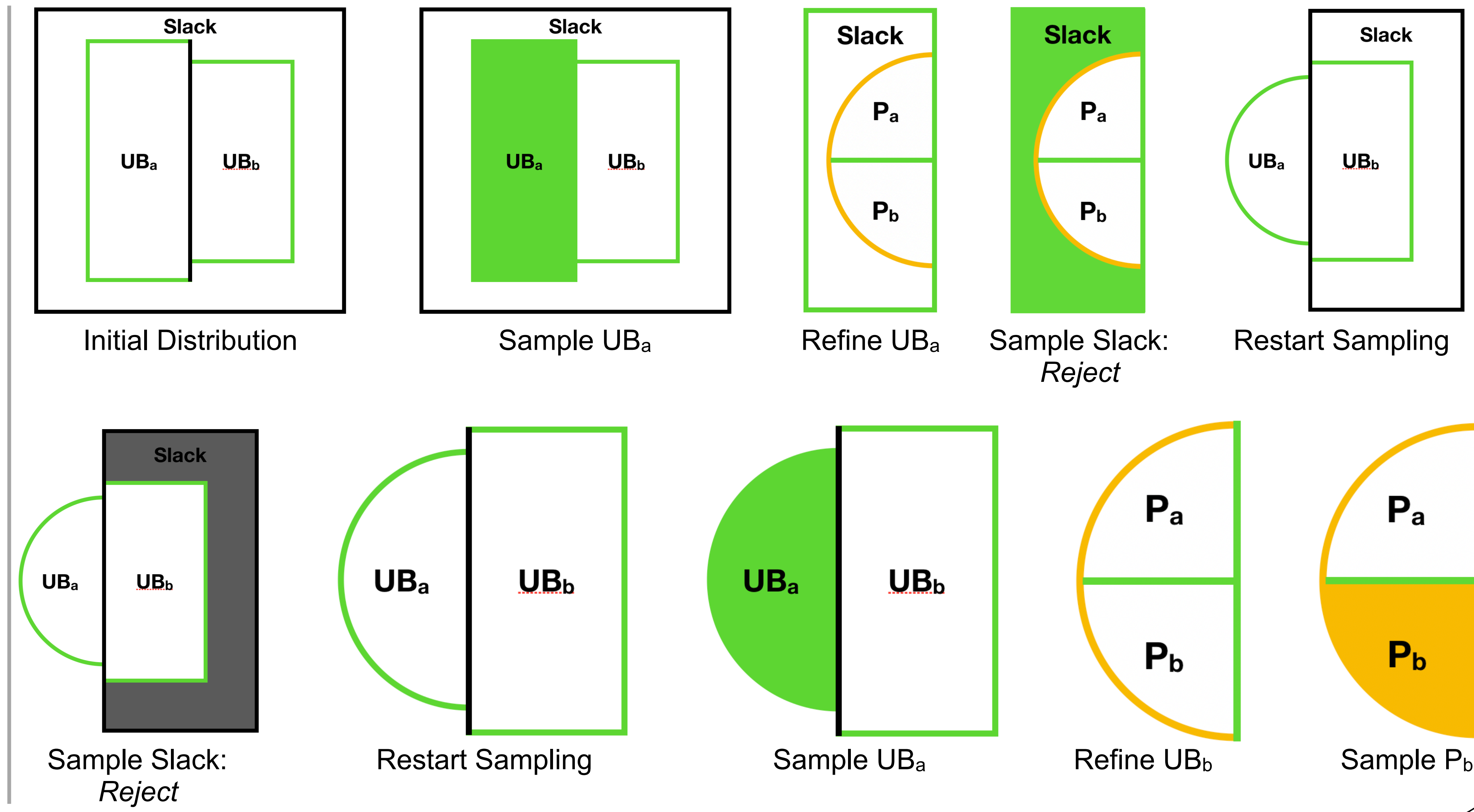
Background: Recursive Rejection Sampling

Rejection sampling provides samples from intractable target distributions using a simpler proposal distribution. The proposal distribution can be defined by upper bounds on the target distribution's partition function that recursively *nest*. Left: the nesting property. Right: the sampling process.



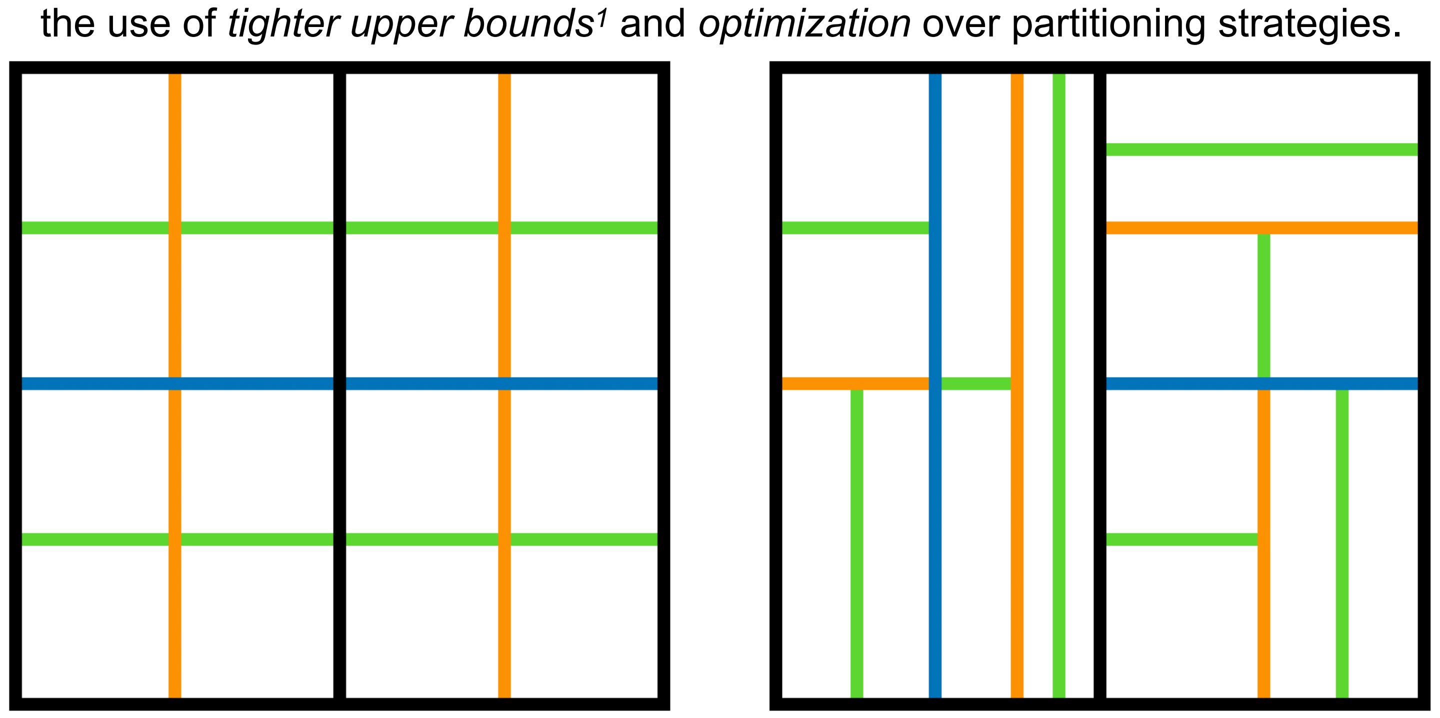
$$p(P_a) = \frac{\mathrm{UB}_a}{\mathrm{UB}} \frac{P_a}{\mathrm{UB}_a} = \frac{P_a}{\mathrm{UB}}$$

The correctness of recursive rejection sampling depends on the upper bounds *nesting*. Here we have a 2d state space with 4 elements. Area corresponds to values of the function representing the unnormalized distribution. After refining the initial upper bound, the sum of UB_a and UB_b must be no larger than the original upper bound (so that the slack is non-negative). We can efficiently nest upper bounds for the matrix permanent.



Adaptive Partitioning

Rather than proving that an upper bound nests according to a predefined partitioning, we adaptively partition based on the specific matrix. This figure loosely represents a 4d state space. On the right, we choose which variable to split the current space with dependent on the previously chosen variables. This allows for the use of *tighter upper bounds*[1] and *optimization* over partitioning strategies.

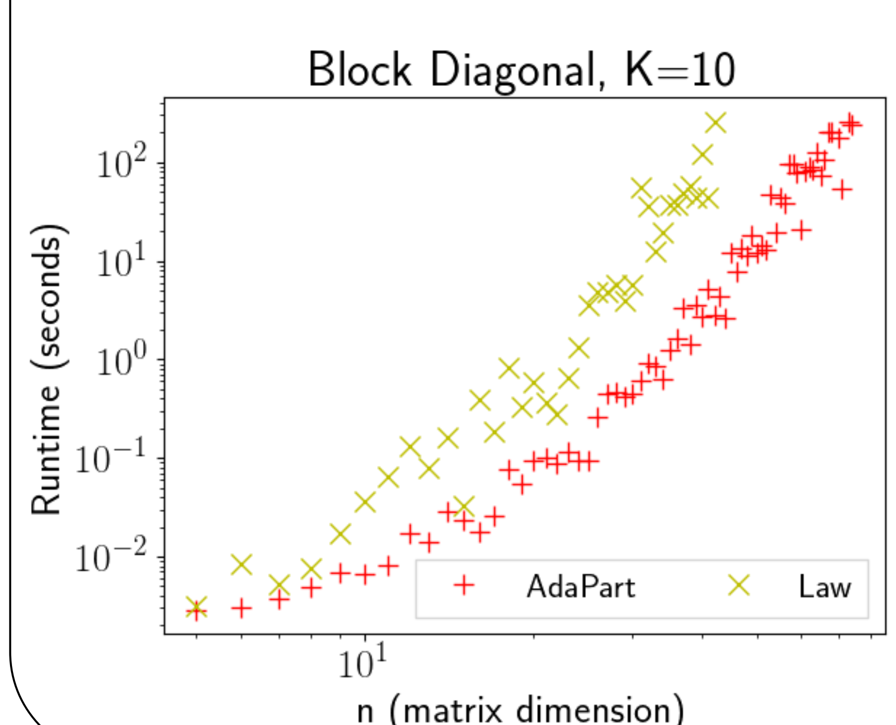


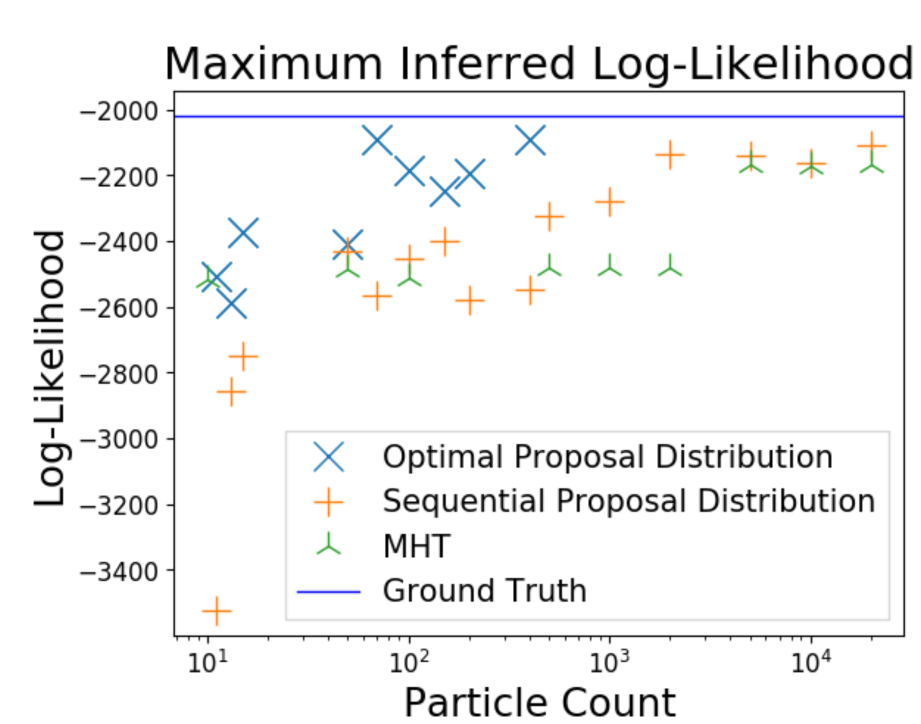
Fixed **Adaptive**

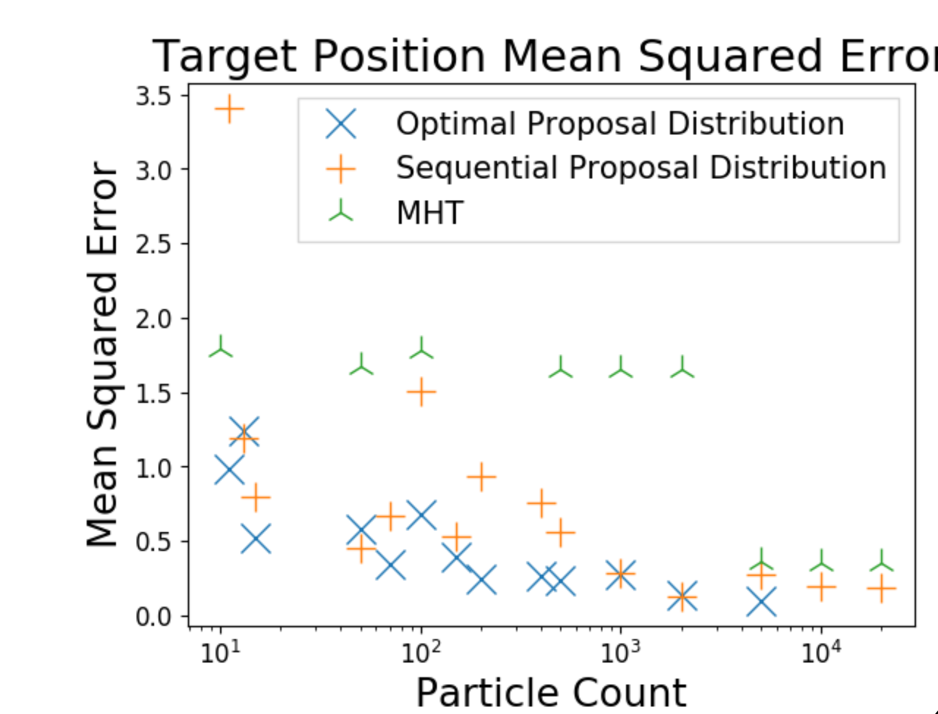
Experiments

Empirical runtime scaling is shown for randomly sampled block diagonal matrices. Our adaptive strategy shows runtime speedups of 10x over prior work[2,3] using a fixed partitioning.

We implemented a Rao-Blackwellized particle filter that uses our algorithm to sample from the optimal proposal distribution and compute approximate importance weights. We compared with a sequential proposal distribution and the multiple hypothesis tracking framework (MHT). On synthetic tracking data we demonstrated that our approach improves tracking performance with an order of magnitude fewer samples.







1. George W Soules. "Permanental bounds for nonnegative matrices via decomposition." *Linear algebra and its applications*, 394:73–89, 2005.
2. Wai Jing Law. "Approximately counting perfect and general matchings in bipartite and general graphs." PhD thesis, Dept. of Mathematics, Duke University, 2009.
3. Mark Huber. "Exact sampling from perfect matchings of dense regular bipartite graphs." *Algorithmica* 2006.