Gaussian Lifted Marginal Filtering

Stefan Lüdtke
Institute of Computer Science
University of Rostock, Germany
stefan.luedtke2@uni-rostock.de

Alejandro Molina
Computer Science Department
TU Darmstadt, Germany
molina@cs.tu-darmstadt.de

Thomas Kirste
Institute of Computer Science
University of Rostock, Germany
thomas.kirste@uni-rostock.de

ABSTRACT

Recently, Lifted Marginal Filtering [5] has been proposed, an approach for efficient probabilistic inference in systems with multiple, (inter-)acting agents and objects (entities). The algorithm achieves its efficiency by performing inference jointly over groups of *similar* entities (i.e. their properties follow the same distribution). In this paper, we explore the case where there are no entities that are directly suitable for grouping. We propose to use methods from Gaussian mixture fitting and merging to identify entity groups and keep the number of groups constant over time. Empirical results suggest that decrease in prediction accuracy is small, while the algorithm runtime decreases significantly.

CCS CONCEPTS

• **Mathematics of computing** → **Sequential Monte Carlo methods**; **Markov processes**;

KEYWORDS

Lifted Marginal Filtering, Multiset rewriting system, Bayesian filtering, lifted probabilistic inference, Gaussian mixture, activity recognition

ACM Reference Format:
Stefan Lüdtke, Alejandro Molina, and Thomas Kirste. 2018. Gaussian Lifted Marginal Filtering. In *Proceedings of 5th international Workshop on Sensor-based Activity Recognition and Interaction (iWOAR'18).* ACM, New York, NY, USA, 3 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

Probabilistic inference in dynamic systems is fundamental for a variety of AI applications. Recently, a novel inference approach (Lifted Marginal Filtering [5]) has been devised, that performs efficient inference in systems with multiple, (inter-) acting agents and objects (entities). In *Lifted Marginal Filtering* (LiMa) [5], states are represented as *multisets* of structured entities (key-value maps). Multisets are used to represent states because this way, identical entities are grouped directly.

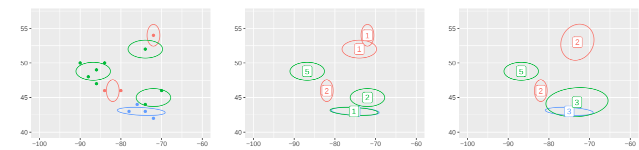

Figure 1: Proposed approach for storing and manipulating the state in the multiplayer game. Different colors denote different players. Left: Actual village positions are shown by dots. We do not use the actual positions, but Gaussian mixture components of village locations, shown by ellipses. Center: Example of state after several conquer actions occurred. Right: New mixture model, after merging of mixture components (to avoid growing the state space indefinitely).

Many Bayesian filtering problems consist of such interacting entities, but often, it is not immediately clear which entities are suitable for grouping – because all of them may have distinct properties. As a running example, consider an online multiplayer game, where multiple players each can have multiple villages. Each village is an entity, consisting of an owner and a location. Each village has a distinct location, therefore no entities are grouped in the multiset. This is problematic when representing distributions $p(S)$ of such multisets s, because we have to enumerate all combinations of village locations).

2 GAUSSIAN LIFTED MARGINAL FILTERING

The key idea of LiMa is to decompose the distribution into two parts: $p(D) = p(T,V) = p(T)\,p(V|T)$. Here, t is the multiset *structure* (how many entities are there, and what are their properties), and v are the property values. This way, in t, we can not only group entities that are identical, but also entities whose values follow the *same distribution* (this procedure is an instance of Rao-Blackwellization [3]). We call a pair of t and a representation of $p(V|t)$ a *lifted state*.

Using these concepts in the multiplayer game, instead of storing the specific location of each village, we describe the location distribution of multiple villages of a player by a

iWOAR'18, September 2018, Berlin, Germany

ACM ISBN 123-4567-24-567/08/06. . . $15.00
https://doi.org/10.475/123_4

normal distribution. The situation depicted in Figure 1 (center) can be represented by the following lifted state[1]:

$$\llbracket\, 2\langle \text{Player: } \mathit{Red}, \text{Pos: } \mathcal{N}(\mu_1, \Sigma_1)\rangle, \ldots$$
$$5\langle \text{Player: } \mathit{Green}, \text{Pos: } \mathcal{N}(\mu_6, \Sigma_6\rangle \,\rrbracket$$

The system dynamics, i.e. the transition model $p(S_t|S_{t-1})$ of Bayesian filtering, is described by a *Probabilistic Parallel Multiset Rewriting System* (PPMRS) [2], which can be computed efficiently using MCMC methods [6].

The important aspect here is to note that the complexity of the prediction step, as well as the cardinality of the posterior state distribution depend only on the number of *different* entities, not on the *multiplicities* of the entities.

The contribution of this paper is to show how such a lifted state representation can be found, in the case when the value distributions are assumed to be Gaussian: In this case, we can fit a Gaussian mixture (using Expactation Maximization [4, 9]) to the properties that we want to represent by a distribution – in the running example, the locations of villages per player. This gives us, for each entity, the mixture component it belongs to, and the parameters of each mixture component. The lifted state is constructed by generating an entity for each mixture component with a value distribution according to the mixture component. The multiplicity of each of those entities in the lifted state is the number of original entities associated with this mixture component. Note that this is an approximation – the idea is that this representation still contains relevant information for prediction.

As an example, consider Figure 1 (left), showing the original village locations, and the estimated mixture components per player. Each component corresponds to a distinct entity in the lifted state (that can each have a multiplicity greater 1).

Another problem is that due to transitions, the number of mixture components increases (see Figure 1, center). This problem is well-known in the context of mixture Kalman filters [1]. For these models, Gaussian mixture model merging methods have been developed: Given a Gaussian mixture, they find a mixture with fewer components that has a minimal distance (in some sense) to the original mixture. Here, we employ the procedure described in [8], an example is shown in Figure 1 (right). By applying this merging operation after each predict step, we reduce the number of distinct entities per player to a fixed number. Thus, inference complexity does not grow over time.

3 EVALUATION

We evaluated the proposed approach on a prediction task in a multiplayer game scenario, which has already been investigated by [10], using a ground state representation.

[1]We use $\langle\cdot\rangle$ to denote entities, i.e. partial functions, and $\llbracket\cdot\rrbracket$ to denote multisets.

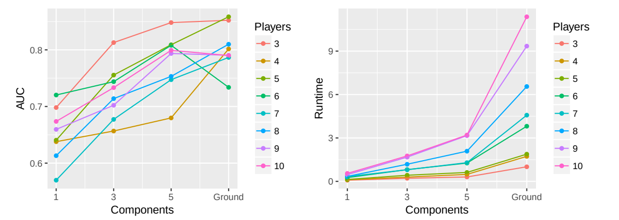

Figure 2: Evaluation results for 1-timestep-prediction. Left: Mean AUC for different number of players and mixture components per player. Right: Runtime for different number of players and mixture components per player.

We predict game state distributions (for varying numbers of players and mixture components per player), and assess the influence of using the lifted state representation on prediction accuracy and runtime using receiver operating characteristic (ROC) curves, and the area under the curve (AUC).

Results are shown in Figure 2. Allowing more mixture components in general leads to a higher AUC. Interestingly, the AUC for 5 components is not significantly lower than the AUC of the ground model ($p > 0.12$ using Wilcoxon signed rank test, $n = 80$), whereas runtime of the lifted models is significantly lower than runtime of the ground model ($p < 10^{-10}$).

4 CONCLUSION

In this paper, we investigated how Lifted Marginal Filtering can be applied to situations without exactly identical entities. The central idea is to estimate a distribution of the properties in which the entities are not identical, and group entities whose property values follow the same distribution. Using an inference algorithm with a complexity that depends only on the number of groups, but not on the number of elements per group, inference can then be much more efficient that when representing each entity individually.

The ideas developed in this paper can be extended in several directions: First, other parametric or non-parametric distributions (instead of normal distributions) could be used for the value distribution $p(V|T)$. Second, a more general approach to identify properties that are suitable to be represented by a distribution is needed. Both of these requirements can potentially be satisfied by using the recently proposed *Mixed Sum-Product Networks* (MSPNs) [7]. MSPNs allow to select properties that are suitable to be approximated by a distribution, and fit non-parametric distributions of the corresponding property values. This would supply us with a very general tool, that does not require any prior knowledge about which property values to approximate by distributions, or about the parametric form of the distribution.

REFERENCES

[1] Daniel Alspach and Harold Sorenson. 1972. Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE transactions on automatic control* 17, 4 (1972), 439–448.

[2] R. Barbuti, F. Levi, P. Milazzo, and G. Scatena. 2011. Maximally Parallel Probabilistic Semantics for Multiset Rewriting. *Fundamenta Informaticae* 112, 1 (2011), 1–17. https://doi.org/10.3233/FI-2011-575

[3] A. Doucet, N. De Freitas, K. Murphy, and S. Russell. 2000. Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 176–183. http://dl.acm.org/citation.cfm?id=2073968

[4] Chris Fraley and Adrian E Raftery. 2002. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association* 97, 458 (2002), 611–631.

[5] Stefan Lüdtke, Max Schröder, Sebastian Bader, Kristian Kersting, and Thomas Kirste. 2018. Lifted Filtering via Exchangeable Decomposition. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*.

[6] Stefan Lüdtke, Max Schröder, and Thomas Kirste. 2018. Approximate Probabilistic Parallel Multiset Rewriting Using MCMC. In *KI 2018: Advances in Artificial Intelligence*.

[7] Alejandro Molina, Antonio Vergari, Nicola Di Mauro, Sriraam Natarajan, Floriana Esposito, and Kristian Kersting. 2018. Mixed Sum-Product Networks: A Deep Architecture for Hybrid Domains. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*.

[8] Andrew R Runnalls. 2007. Kullback-Leibler approach to Gaussian mixture reduction. *IEEE Trans. Aerospace Electron. Systems* 43, 3 (2007).

[9] Luca Scrucca, Michael Fop, T Brendan Murphy, and Adrian E Raftery. 2016. mclust 5: Clustering, classification and density estimation using gaussian finite mixture models. *The R journal* 8, 1 (2016), 289.

[10] I. Thon, N. Landwehr, and L. De Raedt. 2011. Stochastic Relational Processes: Efficient Inference and Applications. *Machine Learning* 82, 2 (2011), 239–272. https://doi.org/10.1007/s10994-010-5213-8