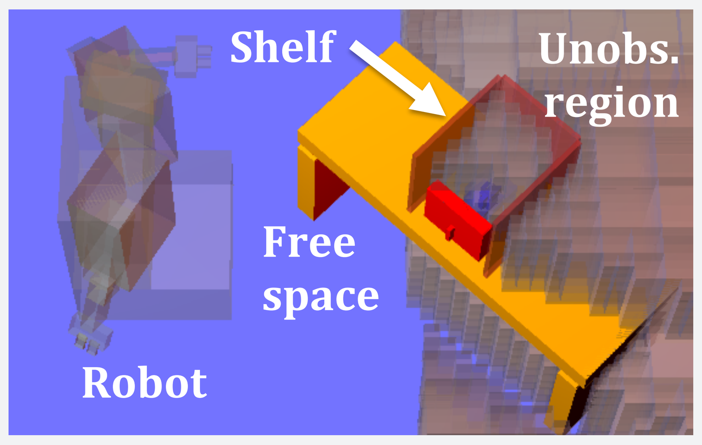
Uncertainty and Estimation in Cyber-Physical Systems: Model Mismatch, Attention, Refinement, and Learning

Lawson L.S. Wong

Learning and Intelligent Systems, Computer Science and Artificial Intelligence Laboratory, MIT

Massachusetts Institute of Technology

Spatial Representation and Estimation on a Mobile-Manipulation Robot



- Understand environment that robot is operating in: Estimate the world state

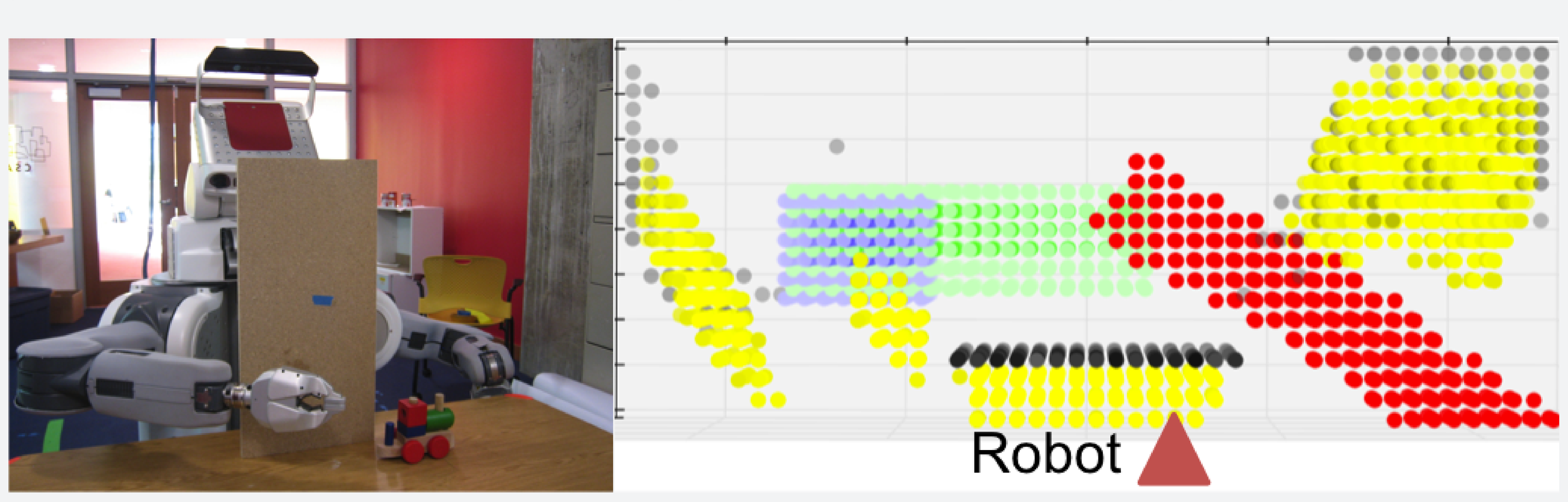
Data Association for Semantic World Modeling from Partial Views

- Previously, I have developed two different estimators for the *world modeling* problem, the estimation of objects' states within the world. Abstractly, on the level of object attributes, a system exists that takes black-box attribute detections, such as object type and pose, and estimates the objects that are present (including their number, which is unknown) and their attribute values.

- This introduces data association issues, because it is unclear which measurements correspond to the same object across different views. I proposed a Bayesian nonparametric batch-clustering approach, inspired by the observation that 'objects' are clusters in joint attribute space. Given attribute detections from multiple viewpoints, this algorithm outputs samples from the distribution over hypotheses of object states, where a hypothesis is a list of objects and their attribute values.

Combining Object and Metric Spatial Information

- Although this gives an elegant 'semantic' view of objects as clusters in joint-attribute space, it ignores crucial information related to the geometric realization of objects, such as their physical extent in space. In particular, low-level observations on whether specific 'voxels' of space are occupied/free cannot be easily incorporated on the object-attribute level. Such observations are traditionally tracked using occupancy grids, and I developed a second estimator that attempts to fuse object-attribute estimates with geometric occupancy grids.

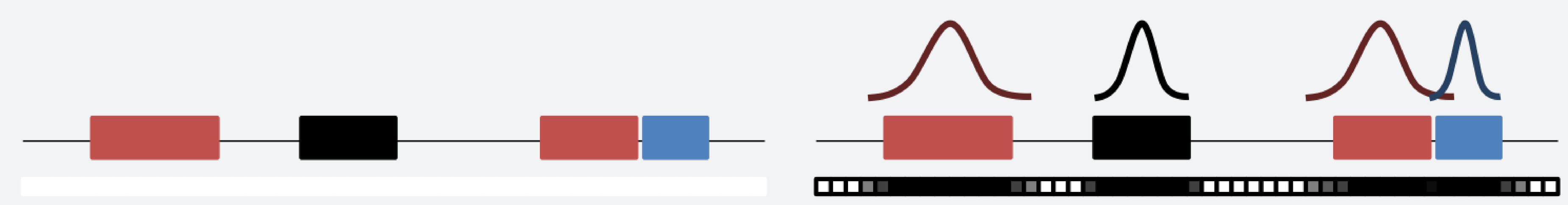


- Because filtering in the joint state involves complex dependencies and is intractable, I instead adopted the strategy of filtering *separately* in the object and metric spaces by using the existing filters. To compensate for the lost dependencies between objects and their geometric realizations, I then developed a way to *merge* the filters on demand as queries about either posterior distribution are made.

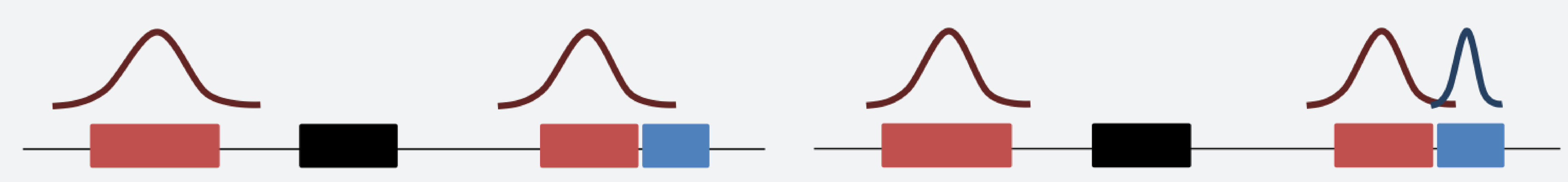
Proposed Framework: Mismatch, Attention, Refinement, and Learning

- **Mismatch: Fault detection.**
  If deviations between expected and observed values exceed thresholds (given by the task), the current model is inadequate, and must be refined.
- **Attention: Task relevance.**
  Without constraints, the model can always be refined until it reaches the model-free case. For many tasks, however, only a small subset of variables benefit the task with additional accuracy. CPS need a way to 'focus' on relevant variables for given tasks.
- **Refinement: Model class expansion.**
  Once a relevant variable's model is inadequate, a larger model class should be explored, *for a small subset of related variables only*.
- **Learning: Estimating parameters.**
  Expanded model classes will have additional parameters to be learned. Ultimately, non-parametric 'models' can act as a final refinement, where empirical estimates are used directly, as in model-free approaches.

Illustration: Model Attention and Selection



- As a proof-of-concept, consider the domain and task above (left). The task is to locate (to some specified uncertainty tolerance) red objects on the real line. The naïve solution is to run all estimators on all the observations, as depicted on the right. Since the task is to locate only red objects, this approach, while sound, is inefficient, especially if the domain is large and contains few red objects.
- Instead, consider the estimator below (left). Only objects whose color attribute is red with high probability are given *attention*; the rest is discarded/ignored. This is conceivably the minimal estimator for the task. However, these observations are very noisy (e.g., the output of an entire object detection pipeline) and lead to large variance in the posterior attribute distribution, above the required tolerance. The performance of this estimator is therefore *mismatched* for the task, and therefore estimator *refinement* is necessary.



- The refinement process involves adding new variables to the estimator and estimating their values based on a buffer of lazily-stored recent observation values. Variables are ranked and added (up to a threshold) if they provide sufficient improvement in expected cost $f(\cdot)$ (in this case, cost = variance):

$$f(p_{X|Y}) \triangleq \mathbb{E}_{y\sim p_Y}\left[f(p_X) - f(p_{X|Y=y})\right] \quad ; \quad p_Y = \int p_{Y|X=x}\, p_X(x)\,\mathrm{d}x$$

- This leads to the addition of two sets of variables. The first set, for the left red object, is a subset of occupancy grid cells; their primary purpose is to distinguish the boundary of the object more finely. The second set, for the right red object, not only does it include associated occupancy grid cells, it also includes the attribute-level variables of the nearby blue object. This latter variable is helpful because of the domain constraint that objects cannot overlap each other, which introduces correlations between the states of the two objects.

Future directions

- Principled mismatch detection: Fault detection & identification / diagnosis
- Human-in-the-loop to provide *interpretable* attention guidance