Anytime, Anywhere Anomaly Recovery through an Online Robot Introspection Framework

Hongmin Wu, Hongbin Lin, Shuangqi Luo, Shuangda Duan, Chen Xiang, Bo Zhao, and Juan Rojas.

Abstract—Robotic introspection and online decision making have been an area of increased focus. The goal is to endow robots with the ability to understand their actions and make timely decisions to reach their goals. Particularly, in unstructured environments, external perturbations are hard to model in low-level control systems and often lead to failure. Robots must then understand nominal and anomalous conditions and trigger timely responses to behaviors that allow the robot to recover and even learn from them and prevent them. Our contribution is the implementation of a fast and robust robot introspection system that allows recovery from (one or multiple) anomalous situations at any point in the task. The system handles both internal modeling errors as well as external perturbations. The robustness of the system is demonstrated across multiple manipulation tasks. The system assumes tasks are decomposed into a sequence of nodes, where each node performs a dual role: one of motion generation and one of introspection. Motion generation is flexible and can be done with any type of accessible approach. Introspection is done by modeling the robots multimodal signals using a range of HMMs including nonparametric Bayesian hidden Markov models. Such models yield strong expressive power to discriminate both nominal and anomalous situations. We made use of a generic strategy for recovery that is easy and flexible to design across different tasks. A new metric for anomaly detection, critical in the proper assessment of the system after recovery has taken place was also designed. We show how the system recovers from both pose estimation errors that lead to collisions in pick tasks as well as external human collisions (which are more likely in co-bot related-tasks). Furthermore, the system is able to robustly recover from collisions that occur at multiple points in the task; even, when anomalies repeatedly appear at a specific point in the task. Supplemental information including videos, code, and result analysis can be found at [1].

I. INTRODUCTION

Human decision making implies awareness. Adult humans are aware of their mistakes and learn to avoid making the same mistake twice. Humans also evaluate whether they have enough information before making a choice and if appropriate to proceed. Their decision’s confidence is correlated with outcome success [2]. In robotics, online decision making and robot introspection have begun to receive more attention recently [3]–[9]. The vision is to endow robots with the ability to understand their actions and make timely decisions to ensure their goals. Particularly, in unstructured environments (where robots are expected to participate increasingly), external perturbations are hard to model in low-level control systems and often lead to failure. Robots must then discern nominal and anomalous conditions (even the types of anomalous conditions) and trigger timely responses to help the robot continue towards its goal or avert failure and recover gracefully. Fig. 1 shows an accidental collision between a human and a robot right at the moment in which a robot picks an object. Normally, this situation would lead to failure, but our system enables the robot to recover and continue its task.

This paper spans the areas of robot introspection, decision making, and anomaly recovery in robot manipulation tasks. In the literature, many works have not attempted an integral approach. Some only model success vs. failure behaviors [3], [6] and others do introspection but not recovery: [4], [8]–[10]. Kappler et al. in [7] present an integrated system for robot introspection with online decision making and anomaly recovery. Furthermore, their system is able to recover even from external perturbations, human collisions into the robot. Their online system has the ability to learn recovery schemes for new errors by asking a human demonstrator to guide the robot back to a nominal state. In their work, however, they perform anomaly recovery at only one stage in a task for a single task and do not provide quantitative analysis of their method (see Sec. II for more details).

Our work studies the feasibility, speed, and robustness of a robot introspection system that can recover from (i) internal model errors or external perturbations such as accidental human collisions at critical moments of different manipulation task in real-time. As human-robot collaboration expands, it’s important not to only guarantee safety for the human user,
but to study whether a robot can continue performing a task after an incident that leaves the robot in an anomalous state. (ii) We also study if the robot can recover, not only at a single incident in the task, but at any point. There seem to be no studies that examine and test anomalous recovery robustly at multiple points in the task. And (iii), we study the robot’s ability not only to recover at multiple points in a task, but also in situations where an external perturbation happens repeatedly at the same point in the task. That is, once the robot recovers, if forced again into an anomalous situation, can the robot recover repeatedly? I.e. can there be a smooth and continuous ability to re-set and re-start a task?

Our contribution is the implementation of a fast and robust robot introspection system that allows recovery from anomalous situations anywhere anytime. The robustness of the system is shown by handling both internal modeling errors and external perturbations across different manipulation tasks.

Tasks are modeled as a directed graphical model. Each node within the graph plays a dual role: it establishes both a robot skill execution, and a trained model for robot introspection (based on nonparametric Bayesian Markov switching processes, see Sec. [IV]). During the execution of every node, the robot introspection system identifies the most likely skill (a specific nominal or an anomalous one if previously learned) or a new anomalous situation (see Sec. [IV-A]). Each node in the task graph is connected to a generic recovery node. In general, we opted for a generic approach given that in unstructured tasks there is a prohibitively large anomaly space, not to mention challenging and computationally expensive. For generic recovery, if an anomaly is experienced during the execution of a skill, a recovery behavior is immediately executed, otherwise a skill continues to completion. Node connections define possible transitions. With regards to the recovery criteria, a node-dependency relationship (see Sec. [V]) determines the degree of roll-back a robot executes in the task before restarting execution. Thus, the introspection system is tightly connected to the lower-level control system, issuing commands to recover or allowing the system to continue. The framework is presented in Fig. 2.

Three manipulation experiments are used to test the robustness of the online decision making system for anomaly recovery. Two experiments (pick-and-place and opening a drawer) test anomalies caused by external perturbations (a human collision) and one experiment (pick-and-place) tests internal modeling errors under different conditions: (i) one anomaly caused during each executed skill and (ii) multiple repeated anomalies triggered during the execution of a skill. F-score, micro- and macro- precision-recall statistics are computed for all experiments and conditions, where F-scores indicates the robot’s recovery rate. For this work, an average recovery rate of 88.18% was achieved for all experiments and conditions. The results indicate strong robustness in this online system, and one that is able to handle anomalies anytime and anywhere. The potential of this system can significantly increase the opportunity of accepting robots in unstructured environments alongside humans. Robots are still prone to frequent failure in unstructured environments. Unless they can gracefully recover in a predictable fashion, it will remain challenging for humans to accept robots as reliable partners. Supplemental information, code, data and videos can be found at [1].

II. LITERATURE REVIEW

The paper spans the areas of robot introspection, failure classification, decision making, and anomaly recovery in robot manipulation tasks. Most works have focused on one of these areas, with few attempting an integrated solution. In [3], Rodriguez et al., designed an “Abort and Retry” solution to the problem of bin-picking using a simple hand. This is an early work that models success vs. failure classification and uses a type of Markov chain to identify a discrete set of moments in which an abort-and-retry attempt should be enacted. The work is limited in a number of ways: it only works at a discrete set of moments; if there is an anomaly, the task must be restarted from the beginning; and it only discerns between success and failure and not between other modes of nominal or anomalous behavior.

In [10], Nakamura et al. present a theoretical error recovery system that works across types of manipulation classes. The paper organizes manipulation tasks hierarchically: with primitives in the bottom, and compound tasks higher up. The work provides recovery solutions through forward and backward correction steps. The latter retry a task by rolling back a certain number of steps (primitive manipulations or compound). The former instead execute minimal adjustment that can be used to finalize a manipulation step. The theoretical work is useful, particularly when there is a well defined
graph of behaviors for a task. However, no experimental work was offered in this work. In [11], Chang et al. devised an error recovery system based on Petri Nets learned from demonstration. Error conditions are defined based on object location: if objects are not located in expected states, an error is triggered. An interesting aspect of the work is that recovery is learned from a human demonstrator. The downside is that the system needs to maintain a growing list of expected object locations. The work does not consider errors that arise from other causes.

All works listed so far assume that anomalies or failure are caused by errors in the internal representation of a robot; namely, sensing, modeling, planning, or execution errors. And while these are certainly relevant, robots now also face a threat from external factors such as unintended collisions with human partners, or the presence and interference of extraneous objects introduced into a robot’s workspace, amongst others.

In [4], DiLello et al. used a non-parametric Bayesian Hidden Markov Model in an alignment task to identify specific failure modes when extraneous objects where placed in the workspace preventing the robot to achieve a proper alignment. His work showed the identification of failure modes using wrench signals. The work however did not attempt recovery measures. In our work, we also use non-parametric Bayesian techniques with multi-modal signals, but in our case, we develop robust recovery techniques across tasks and conditions. The work of Park et al. in [8] studied the effects of multi-modal sensory signatures in a hidden Markov model (HMM) for anomaly identification. Their work identified anomalies in pushing tasks (doors and switches) and feeding tasks. The anomaly threshold was updated according to the progress of the task, but the work did not test any kind of recovery. As in this work, we too use Bayesian priors, but we make use of a non-parametric form that allows us to learn the complexity of each mode according to the data, allowing us to generate more expressive identification models which directly affects our task recognition and recovery rates [12]. The work of Salazar et al. in [13] introduced anomaly recovery by using human mind signals in real-time to alert the robot if it had made a mistake. The work used EGG Error-Related Potential (ErrP) signals as well as secondary interactive error-related potential signals that further alerted the robot if the human caught a second mistake. The approach is compelling as the human is able to influence the robot’s behavior but also the robot influences the human behavior. The work did not study how to help the robot learn from experience. One of our goals is to grow a library of motion/identification models that the robot accumulates over time to learn new behaviors.

Finally, the work in [7] devices a supervised machine-learning framework for online decision making in manipulation tasks. The system closes a loop between a high-level decision making system with a low-level loop. The high-level system makes use of two classifiers to identify nominal behaviors and failure. The system learns new skills online, including recovery skills and is able to save them as Dynamic Motion Primitives (DMPs) in the low-level layer. This work advanced the state-of-the-art significantly by integrating robot introspection, failure characterization, decision making, and anomaly recovery. However, the work did not provide quantitative results for recovery and only showed one recovery for one task at one moment in the task. We are interested in further studying the robustness of recovery behaviors. That is, can a system recover multiple times from failure? And it can do so anywhere throughout the task? Can it do so across multiple tasks? Can it withstand both failure caused by internal errors and by external perturbations? Our goal is effective recovery anytime anywhere.

III. PROBLEM FORMULATION

Several researchers, including our previous work, understand motion to posses an inherent structure composed of a sequence of primitive or compound skills. Motion’s structure is compared to that of grammars [7], [9], [14], [15]. That is, just as grammar has a set or rules and order to organize works, so motion to is organized by a set of rules and order that give way to discernible patterns in the sensory-motor action space. Based on this premise, we use a directed graphical model composed of nodes with dual roles to define both motion generation and motion identification instructions. Any task graph is bootstrapped by a simple linear structure that can grow as more skills and identification models are learned over time. With respect to motion generation, we do not limit the form in which motion is devised (smooth joint-interpolation, motion planning, or parametrized motions like Dynamic Movement Primitives or Probabilistic Movement Primitives [16]). With respect to motion identification, we use trained models derived from a variety of Bayesian non-parametric Markov (switching) models presented in [12] (see Sec. [15] for more details). Fig. 3 shows an illustration of a library containing a set of task modules for motion generation and motion identification, both of which are called simultaneously by nodes in the graph. During the execution of a node, robot introspection uses multi-modal robot sensory
inputs to identify the most likely running skill. The system works as an arbiter: if a nominal skill is identified, a green light gives the system the ability to continue to the next skill in the graph. If an anomaly is detected (see Sec. IV-A for details), a red light impedes normal progression and triggers a recovery behavior. With regards to anomaly recovery, it is possible design the framework to allow for generic recovery or alternatively to for specific failure recovery. In structured environments, learning a significant portion of failure-modes solutions a priori might be conceivable, though still costly in time. However, we think that in unstructured environments the number of possible failures is prohibitively large. For this reason, we opted for bootstrapping the system with 

IV. ROBOT INTROSPECTION

The robot introspection model is tested on a variety of algorithms that use a non-parametric Bayesian priors along with a Hidden Markov model (HMM) and either Gaussian or Autoregressive emission models. HMMs are a stochastic and generative process used to make inference on temporal data. HMMs contain a finite and fixed number of latent states or modes which generate observations through mode-specific emission distributions. Transition distributions control the probability of transitions across time steps. The model assumes (discrete or continuous) conditionally independent observations given the generating latent state. Given a set of observations, the forwards-backwards algorithm is used to infer model parameters. Bayesian nonparametric priors extend the HMM model to learn the complexity of the HMM according to the data. In particular, we study HMMs that use the sticky-Hierarchical Dirichlet Process (sHDP) prior. A detailed mathematical presentation for the sHDP-HMM family of algorithms is found in [12]. A brief presentation is only offered here.

The sHDP-AR-HMM uses a Hierarchical Dirichlet process prior over the transition distribution of an HMM and uses Bayes to learn the number of (potentially infinite) modes. The prior also employs a sticky parameter that discourages fast-mode-switching otherwise present. Additionally, the use of an autoregressive suggests temporal dependencies in the observations. The graphical model uses a time-series \{y_1, y_2, ..., y_T\} of observed multi-modal data, a matrix of regression coefficients \(A^{(k)} = [A_1^{(k)}, \ldots, A_d^{(k)}] \in \mathbb{R}^{d \times (d+r)}\) with \(n\) dimensions \(d\) and \(r\)th order autoregressive, and a measurement noise \(\Sigma\), with a symmetric positive-definite covariance matrix. The generative model for the sHDP-AR-HMM is summarized as:

\[
G_0 = \sum_{k=1}^{\infty} \beta_k \delta_{\theta_k}, \quad \beta|\gamma \sim GEM(\gamma), \\
\theta_k|G_0 \sim G_0, \\
G_j = \sum_{k=1}^{\infty} \pi_{jk} \delta_{\theta_k}, \quad \pi_{jk}|\alpha, \beta \sim DP(\alpha, \beta), \\
\pi_0^{(i)} \sim \pi_1^{(i)}, \\
y_t = \sum_{i=1}^{r} A_i^{(z_t)} y_{t-i} + e_t(z_t), \quad e_t \sim \mathcal{N}(0, \Sigma).
\]

Where, \(G_0\), is a global probability measure and is the result of the product of weights \(\beta_k\), which are sampled via stick-breaking process and draws from the base measure \(H\) which is a Dirichlet Process. \(G_0\) is then used to to define a prior on the set of HMM transition probability measures \(G_j\). The transition probability vector \(\pi_j\) is drawn from a Dirichlet process from the mode at the previous time step. With an autoregressive model, the observations are a linear combination of the past \(r\) mode-dependent observations with additive white noise. For Gaussian models, mode specific means and standard deviations are used \(\theta_{z_t} = \mathcal{N}(\mu, \sigma^2)\).

A. Identification

In our robot introspection system, we simultaneously detect nominal skills and anomalous unexpected events. We train our model on individual skills and capture their dynamics through a multi-modal observation vector \(\tau_m\) consisting of end-effector pose and wrench values. Scalable incremental or “memoized” variational coordinate ascent, with birth and merge moves [17] is used to learn the posterior distribution of the sHDP-HMM along with mean values for the model parameters \(\Pi\) of a given skill \(s\). Hence \(\Pi_s = \{\pi, A\}\) the transition matrix and regressor coefficients.

1) Nominal Classification: Given \(S\) trained models for \(M\) robot skills, scoring is used to compute the expected cumulative likelihood of a sequence of observations \(E[\log P(Y|\Pi_s)]\) for each trained model \(s \in S\). Given a test trial \(r\), the cumulative log-likelihood is computed for test trial observations conditioned on all available trained skill model parameters \(\log P(y_{r_1:r_i}|\Pi_s)\) at a rate of 200Hz (see Fig. 4 for an illustration). The process is repeated when a new skill \(m\) is started. Given the position in the graph \(s_c\), we can index the correct log-likelihood \(I(\Pi_s = s_c)\) and see if its probability density of the test trial given the correct model is greater than the rest:

\[
\log P(y_{r_1:r_i}|\Pi_{\text{correct}}) > \log P(y_{r_1:r_i}|\Pi_s)v \quad \forall s(s \in S \land s \neq s_c).
\]

If so, the identification is deemed correct, and the time required to achieve the correct classification recorded. At the end of the cross-validation period, a classification accuracy matrix is derived as well as the mean time threshold value
(these results were reported in [12], in this paper we limit ourselves to report on the recovery robustness of the system).

2) Anomalous Classification: Anomaly detection is based on the notion that during nominal tasks the cumulative likelihood has similar patterns across trials of the same robot skill. If so, the expected cumulative log-likelihood $L$ derived in training can be used to implement an anomaly threshold $F$. Initially, we consider a likelihood curve $L$ generated from training data for a given skill $s$. Then, for each time step in an indexed skill $s_{c}$, the anomaly threshold is set to $F1_{s_{c}} = \mu(L) - k \times \sigma(L)$, where $k$ is a real-valued constant that is multiplied by the standard deviation to change the threshold. Here, we are only interested in the lower (negative) bound. Then, an anomaly is flagged if the cumulative likelihood crosses the threshold at any time: if $log P(y_{t:T} \mid \Pi_{\text{correct}}) < F1_{s_{c}}$: anomaly, else nominal. In Fig. 4 note the 4 probabilistic models, and given an indexed position in the graph, one can derive the anomaly threshold that corresponds to that skills expected cumulative log likelihood. The figure also illustrates how at the end of a skill, all data is reset and restarted.

Upon our initial exploration of recovery schemes we noticed that after resetting the cumulative log-likelihood observations in the HMM model, anomalies were immediately triggered at the beginning of the skill. Further examination revealed that the standard deviation of cumulative log-likelihood graphs during training began with small variances but then these grew over time (Fig 5 illustrates this effect). Given that variances are small at the beginning of the task, even small variations in observations can easily trigger cross-thresholding. A second threshold definition was designed to overcome this situation. As the difference between $L$ and $F1$ is minimal at first the new anomaly threshold $F2$ (for an indexed skill) instead is focused on computing the derivative of the difference: $F2_{s_{c}} = \frac{d[L-F1_{s_{c}}]}{dt}$. Fig. 6 shows an example of how the derivative signal crosses the empirical threshold multiple times as anomalies are triggered by external perturbations during the execution of a skill.

V. ANOMALY RECOVERY

As stated earlier, anomaly recovery, from the outset is designed to be generic such that one can easily and flexibly design the recovery scheme for any given task. Given the large number of potential tasks, the even larger number of possible anomalies (not to mention doing this across robots) designing specific recovery skills a priori is exceptionally hard. Instead, learning error modes and corresponding recovery skills over time seems more practical and in-line with the way biological systems learn.

Our generic recovery scheme is designed to roll-back to a stable point in the task and retry to execute the behavior while continuing to track the target goal. An underlying question to the design is how far back should we roll-back? In this work, we limit our roll-back to the beginning of a given skill. We ask, should we repeat the same skill or reattempt skills further back in the chain? This question relates to the work of Nakamura et al. in [10]. However, in our work, we use a state-dependency criteria: that identifies whether the current skill depends on the previous one for correct execution. Where, correct execution is defined as giving the robot the right pose to complete the skill once again. Dependencies are currently annotated manually. The system is recursive. Once a recovery flag is triggered, we look back until no more dependencies exist. A simple illustration is offered: Grasping or picking skills, usually require a pre-grasp or pre-pick pose to facilitate the goal. If a pick action fails, due to a perturbation. Re-attempting the skill directly would likely fail as the end-effector pose or target object location would have been modified. The system identifies that the pick/grasp node has a dependency and it rolls back to the previous skill where the end-effector is set to a pre-grasp/pick pose. The system again analyzes if that skill has a
dependency, which in this examples does not. The roll-back halts, and the system resumes the task from that point on.

The anomaly detection system acts as an arbiter. If no anomaly is identified the skill continues executing until its termination and transitions to the next skill. If an anomaly is present, we roll-back. The introspection system is tightly connected to the lower-level system, issuing commands to control the execution of the robot. Finally, note that during the recovery stage in which the manipulator returns to the pose of a previous node, the robot introspection is shut-down. Under this assumption, the system is not able to identify perturbations while undergoing recovery. This is left as future work.

VI. EXPERIMENTS AND RESULTS

Three manipulation experiments are designed to test the robustness of the online decision making system for anomaly recovery. Two tasks (pick-and-place and open-and-close-a-drawer) test anomalies caused by external perturbations (a human collision) and one experiment (pick-and-place) tests internal modeling errors. For each of the three experiments, we test four separate conditions: (i) no anomalies, (ii) anomalies without recovery, (iii) one anomaly caused at each executed skill and (iv) multiple anomalies caused at each skill. The pick-and-place task consists of 5 basic nodes (not counting go-to-home and the recovery node): go-to-home, go-to-pick-hover, go-to-pick, go-to-pick-hover, go-to-place-hover, and go-to-place. The open-and-close-a-drawer task consists of 5 nodes: go-to-grip-hover, grip, pull-to-open, push-to-close, and go-back-to-start. A dual-armed humanoid robot -Baxter- was used. All code was run in ROS Indigo and Linux Ubuntu 14.04 on a mobile workstation with an Intel Xeon processor, 16GB RAM, and 8 cores. In terms of motion generation, two techniques were used for the different tasks. For the pick-and-place task, we used Baxter’s internal joint trajectory action server which uses cubic splines for interpolation. The goal target is identified online through image-processing routines. For the open-and-close-a-drawer task, we trained dynamic movement primitives using kinesthetic teaching.

In terms of robot introspection, the sHDP-HMM code with “memoization” variational coordinate ascent, with birth and merge moves was implemented using bnpy [18] wrapped in ROS. Batches of 10 trials were used at a time for training. The observations consisted of a 13 dimensional vector consisting of Baxter’s 7 end-effector pose values (position and quaternion) and 6 wrench values. A baseline HMM algorithm was implemented through HMMLearn [19] and wrapped with ROS. The anomaly threshold for each skill was computed through leave-one-out cross-validation. External perturbations were generated by human collisions. The perturbations were strong enough to cause Baxter’s compliant arms to move significantly from its target goal and sometimes collide with other parts of the environment.

Fig. 4 shows a representative image of the Baxter robot attempting a pick operation. A human collaborator accidentally collides with robot before a pick action. The robot introspection system identifies an anomaly and triggers a recovery behavior. The lower left part of the image shows the anomaly F2-metric indicating that an anomaly occurred. The system then flags a recovery transition, which is seen in the directed graph on the right (and was implemented in ROS-SMACH). Video for the three experiments, under the four conditions, for both of the introspection models are all available in our project website, along with code, and additional result details [1]. For the results reporting, we use F-score precision recall graphs for the three experiments, under the latter two conditions: (iii) one anomaly caused at each executed skill and (iv) multiple anomalies caused at each skill. The first two conditions are not reported as they do not provide insights into robustness. Their purpose is to provide a baseline contrast with respect to the generic recovery behavior. In summary, the system shows a very high level of robustness.

1) External Perturbations: For the case where there is one anomaly per node (7 afflicted anomalies for pick-and-place while 5 for open-and-close-a-drawer) the pick-and-place task ran 27 test trials while the drawer task ran 24 test trials. The pick-and-place recovery rate was 88% with our baseline method, a bit smaller with the sHDP-HMM method, but nonetheless a significant recovery rate was achieved. The precision was 100% (both for macro/micro) indicating strong resistance to false positives. The recall was ~82% (for macro/micro) indicating the existence of some false-negatives. The drawer task recovered 91.67% of the time. Precision was 95% (macro/micro) and recall was ~84.5%. The system showed better recovery performance in the presence of multiple anomalies compared to a single one per node.

For the case of multiple anomalies per node, the pick-and-place task ran 42 test trials while the drawer task ran 30 test trials. The pick-and-place recovered 85% of the time with recall ~84.5% (micro/macro) and precision ~95%. For the
drawer task, we recovered 93.3% of the time, with recall \( \sim 93\% \) (micro/macro) and precision \( \sim 100\% \).

2) Internal Modeling Errors: The internal modeling error experiment recovery only ran on the pick-and-place task. For this task, the robot recovered \( \sim 88.89\% \) for the time from collisions caused by inaccurate estimation of the target object’s pose. All the results are tabulated in Table 1.

VII. DISCUSSION

Our work demonstrated anytime, anywhere anomaly recovery through a fast, feasible, and robust robot introspection system. It did so under significant anomalous conditions – human collisions as a source of external perturbations and manual objects displacements. We showed that a robot could recover from external and internal errors, as well as when anomalies were present at every node/skill in a task; and even in more harsh conditions, where multiple anomalies (we tested five consecutive ones) occur back-to-back at each node. In the latter case, which is best appreciated in the videos [1], the robot seems to be in constant duress and yet recovers consistently and repeatedly. Overall, our recovery rates ranged from 85% to 93.3% with an average of 88.18% for all three experiments and the different conditions.

It seems to the authors that there are not many works in the literature that have explored the subject of recovery with real robots in unstructured environments under the presence of significant external perturbations. Furthermore, it seems that by using a more expressive model to do robot introspection, our recovery ability also increased. We will continue to explore improved models that can better capture the spatio-temporal relationships of high-dimensional multi-modal data. As well as looking for representations that scale over time in order to acquire a useful and practical library of skill identification and motion generation.

Yet there is much work to be done. In our quest to approach human-level insight, it is crucial that we develop robust metrics for action-confidence. How sure am I that I am working nominally, or that I am in a new and potentially dangerous situation? Currently our thresholds are derived empirically, we will seek to learn them. Furthermore, as human adults learn from their mistakes, we would like to help the robot learn from the past and further learn to prevent mistakes by anticipating the consequences. With respect to the roll-back in recovery, we currently roll-back a given

| TABLE I |
| THIS TABLE SHOWS RESULTS FOR 3 EXPERIMENTS WITH TWO TYPES OF ANOMALY CAUSES: EXTERNAL PERTURBATIONS AND INTERNAL ERRORS. WE ALSO COMPARE THE ROBUSTNESS OF THE RECOVERY BEHAVIOR IN THE PRESENCE OF 1 ANOMALY PER SKILL AND MULTIPLE ANOMALIES PER SKILL IN A TASK. ADDITIONALLY, WE COMPARE THE PERFORMANCE OF THE MORE EXPRESSIVE sHDP-HMM MODEL WITH AN HMM THAT SERVES AS A BASELINE. F-SCORE, RECALL, AND PRECISION STATISTICS ARE PROVIDED. |

|                  | HMM   | sHDP-HMM |
|------------------|-------|----------|
|                  | Micro | Macro    | Micro | Macro    | Micro | Macro    |
|                  | F     | Recall   | Precision | Recall   | Precision | F     | Recall   | Precision |
| External Pert.   |       |          |          |         |          |       |          |          |
| Pick & Place     |       |          |          |         |          |       |          |          |
| 1 anomaly / skill | 88.00%| 88.00%   | 95.65% | 88.00%   | 96.67% |        | $2.00\%$| 81.48%   | 100%     | 82.00%   | 100%     |
| 1 mult. anomaly / skill | 84.06%| 82.93%   | 97.14% | 83.50%   | 97.14% | $8.50\%$| 84.09%   | 94.87%   | 85.06%   | 95.28%   |
| Open Drawer      |       |          |          |         |          |       |          |          |
| 1 anomaly / skill | 80.00%| 80.00%   | 100%   | 80.00%   | 100%   |        | 91.67%  | 90.00%   | 81.82%   | 90.00%   | 85.33%   |
| 1 mult. anomaly / skill | 72.34%| 71.79%   | 100%   | 72.34%   | 100%   |        | 93.33%  | 93.33%   | 100.00%  | 93.33%   | 100.00%  |
| Internal Pert.   |       |          |          |         |          |       |          |          |
| Pick & Place     | 71.43%| 71.43%   | 100%   | 71.43%   | 100%   |        | $8.89\%$| 100%     | 88.89%   | 100%     | 88.89%   |
number of skills. This may be too large of a step for some situations. We would like to explore identifying latent-states within an HMM models that might better serve as point-of-return and increase efficiency in the recovery procedures.

VIII. CONCLUSION

In this work we presented an Anytime, Anywhere Anomaly Recovery Online Robot Introspection framework for Decision Making. We introduced a system by which tasks are decomposed into a sequence of nodes and where each node performs a dual role: one of motion generation and one of introspection. Motion generation is flexible and can be done with any type of accessible approach. Introspection is done by modeling the robot’s multi-modal signals using different HMM variants including nonparametric Bayesian hidden Markov models. Such models yield strong expressive power to discriminate both nominal and anomalous situations. We made use of a generic strategy for recovery that is easy and flexible to design across multiple tasks. We also made use of a new metric for anomaly detection, critical in the proper assessment of the system after recovery has taken place. Furthermore, we showed that our system recovers from both inner model errors or external perturbations, like collisions from humans, which are more likely in upcoming co-bot related-technologies. Not only so, the system was able to robustly recover from multiple significant anomalies across the duration of the task, and from multiple-repeated anomalies occurring at multiple places in the task.

IX. ACKNOWLEDGEMENTS

This work is supported by “Major Project of the Guangdong Province Department for Science and Technology (2014B090919002), (2016B0911006).”

REFERENCES

[1] J. Rojas, “Supplement to online robot introspection for decision making in anomaly recovery,” 2017. [Online]. Available: http://www.juanrosas.net/statdep_recovery/.
[2] N. Yeung and C. Summerfield, “Shared mechanisms for confidence judgements and error detection in human decision making,” in The cognitive neuroscience of metacognition. Springer, 2014, pp. 147–167.
[3] A. Rodriguez, M. T. Mason, S. S. Srinivasa, M. Bernstein, and A. Zirbel, “Abort and retry in grasping,” in Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on. IEEE, 2011, pp. 1804–1810.
[4] E. Di Lello, M. Klotzbucher, T. De Laet, and H. Bruyninckx, “Bayesian time-series models for continuous fault detection and recognition in industrial robotic tasks,” in Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE, 2013, pp. 5827–5833.
[5] O. Kroemer, H. Van Hoof, G. Neumann, and J. Peters, “Learning to predict phases of manipulation tasks as hidden states,” in Robotics and Automation (ICRA), 2014 IEEE International Conference on. IEEE, 2014, pp. 4009–4014.
[6] J. Rojas, K. Harada, H. Onda, N. Yamanobe, E. Yoshida, and K. Nagata, “Early failure characterization of cantilever snap assemblies using the pa-reclit,” in IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 3370–3377.
[7] D. Kappler, P. Pastor, M. Kalakrishnan, M. Wuthrich, and S. Schaal, “Data-driven online decision making for autonomous manipulation,” in Proceedings of Robotics: Science and Systems, Rome, Italy, 2015.
[8] D. Park, Z. Erickson, T. Bhattacharjee, and C. C. Kemp, “Multimodal execution monitoring for anomaly detection during robot manipulation,” in 2016 IEEE International Conference on Robotics and Automation (ICRA), May 2016, pp. 407–414.
[9] J. Rojas, S. Luo, D. Zhu, Y. Du, H. Lin, Z. Huang, W. Kuang, and K. Harada, “Online robot introspection via wrench-based action grammars,” to appear in iros 2017. preprint arXiv:1702.08695, 2017.
[10] A. Nakamura, K. Nagata, K. Harada, N. Yamanobe, T. Tsuji, T. Foisette, and Y. Kawai, “Error recovery using task stratification and error classification for manipulation robots in various fields,” in Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE, 2013, pp. 3535–3542.
[11] G. Chang and D. Kulic, “Robot task error recovery using petri nets learned from demonstration,” in Advanced Robotics (ICAR), 2013 16th International Conference on. IEEE, 2013, pp. 1–6.
[12] J. Rojas, H. Wu, and K. Harada, “Robot introspection with bayesian nonparametric vector autoregressive hidden markov models,” preprint arXiv:1705.08661, 2017.
[13] A. F. Salazar-Gomez, J. DePretto, S. Gil, F. H. Guenther, and D. Rus, “Correcting robot mistakes in real time using eeg signals,” ICRA. IEEE, 2017.
[14] H. C. Lin, I. Shafarian, D. Yuh, and G. D. Hager, “Towards automatic skill evaluation: Detection and segmentation of robot-assisted surgical motions,” Computer Aided Surgery, vol. 11, no. 5, pp. 220–230, 2006.
[15] J. Rosen, J. D. Brown, L. Chang, M. N. Sinanan, and B. Hannaford, “Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete markov model,” Biomedical Engineering, IEEE Transactions on, vol. 53, no. 3, pp. 399–413, 2006.
[16] F. Meier and S. Schaal, “A probabilistic representation for dynamic movement primitives,” arXiv preprint arXiv:1612.05932, 2016.
[17] E. B. Fox, M. C. Hughes, E. B. Sudderth, M. I. Jordan et al., “Joint modeling of multiple time series via the beta process with application to motion capture segmentation,” The Annals of Applied Statistics, vol. 8, no. 3, pp. 1281–1313, 2014.
[18] bnpy, “Bayesian nonparametric machine learning for python,” 2017. [Online]. Available: https://github.com/bnpy/bnpy/.
[19] hmmlearn, “Hidden markov models in python,” 2017. [Online]. Available: https://github.com/hmmlearn/hmmlearn/.