A Relative-Change-Based Hierarchical Taxonomy for Cantilever-Snap Assembly Verification.

Intelligent Sys. Research Institute, AIST
Tsukuba, Ibaraki, 305-8568, Japan

Abstract—Snap assembly automation remains a challenging task. While progress is being made in localization of parts, force controllers, and control strategies, little work has been done to help the robot reason about its current state, such that if necessary, the robot can assume corrective actions to accomplish the task. Error prone situations caused by the unexpected motion of parts, localization errors, jamming or wedging, cannot be solved by force controllers alone. For this reason we propose a snap assemblies verification system for cantilever-snap fasteners. The verification works in concert with a control strategy that makes use of constraint designs embedded in the snap parts’ physical design. The constrained assembly motion generates similar sensory-signal patterns across trials that facilitates force signal discrimination into higher level abstractions of intuitive behavior. This work’s contribution is the design of a hierarchical taxonomy for cantilever-snap verification based on increasingly abstract layers that encode relative-change in the task’s force signatures. A five-layered taxonomy is built on the concept that relative-change patterns can be classified through a small category set and aided by contextual information. The verification system yielded human apropos intuitive categorizations of task behavior for every state and effectively determined the assembly result. This simple yet effective approach will be expanded to perform probabilistic online system verification to aid in fault tolerance and the automation of cantilever-based snap assemblies.

I. INTRODUCTION

The snap assembly automation process still presents many challenges and remains largely a manual industrial task. To date, parts localization [1], [2], force controller implementations [3], control strategies [4], and contact monitoring [5], [6] are an open area of research and provide valuable steps in automating the assembly process. However, assembly automation is not possible if the robot cannot respond to unexpected events. Localization of snap fasteners is challenging and all the more in the presence of multiple snaps. Force controllers alone cannot respond to unforeseen events like wedging and jamming situations, or those in which parts move unexpectedly during assembly. Snap parts offer more challenges as they consist of movable elastic parts that can wear off. Snap parts, such as those in cell phones or cameras, also consist of internal hardware lining designed to facilitate insertions, however, when parts alignment is not optimal, the hardware contributes to jamming phenomena. A largely open area or research in automating the snap assembly problem is enabling the robot to reason about its actions in order to take corrective measures.

To this end, a robot must monitor the assembly process to have a high-level understanding of its state, and if necessary, respond specifically to correct a motion, or re-attempt it completely. In this work, we devised a system that acquired a high-level understanding of the robot’s behaviors throughout the snap assembly. To facilitate the interpretations of force signals into high level behaviors, two conditions were considered. The first was that snap assemblies must be classified according to one of three snap fastener classes: the cantilever snap, the annular snap, and the torsional snap class [7]. The cantilever class is the most common and the one we examined. The second condition requires a control strategy that generalizes across various degrees of geometrical complexity within a snap class and a control strategy that yields consistently reproducible sensory-motor signals across trials. Both requirements enable the generation of force-signals whose consistent patterns across trials facilitate the discrimination and clustering of signals, and hence their diagnosis.

Our current research builds on prior work which designed the “Pivot Approach” control strategy [4]. The latter strategy made use of constraint designs embedded in the snap parts’ hardware and constrained the assembly’s motion by pivoting about a docking point common to cantilever class parts. The Pivot Approach systematically discretized the cantilever snap assembly into five intuitive states: Approach, Rotation, Alignment, Snap, and Mating. Snapshots of the process can be visualized in Fig. 1.

We now study the implementation of a process monitoring system that hierarchically abstracts useful assembly performance information at each stage to determine: i) executed behaviors, ii) behavior sequence, and iii) if the task was a success. This work’s contribution is the design of a hierarchical taxonomy for cantilever-snap verification based on increasingly abstract layers that encode relative-change in the task’s force signatures. The five-layered taxonomy is built on the concept that patterns of relative-change can be classified through a small set of labels and aided by contextual information. The first layer uses linear regression to approximate each of the six dimensional force-moment signals and discretize the data in linear segments. Each segment is considered a primitive whose gradient, amplitude, average value, and time statistics are encoded in a data structure and classified according to 9 possible gradient categories. The next layer of abstraction encodes primitives into seven basic motion compositions by looking at the ordered sequence of primitives. Similarly, the third layer
abstracts low-level behaviors (LLB’s) by looking at the ordered sequence of composites over a small window. The fourth layer contextualizes the process monitoring by looking at key LLB’s that occur in each state and determines whether they successfully represent, or not, the human-apropos high-level behaviors (HLB’s) of Approach, Rotation, Alignment, Snap Insertion, and Mating. The fifth layer examines the sequence of HLB’s if the task was successful or not. Currently the system runs offline and aims at properly building the verification system.

The verification system was ran on simulation signals yielded by a PA-10 industrial robot running the Pivot Approach to perform the assembly of a camera consisting of a male and female part with cantilever-snaps and internal ridging. The system’s output correctly interpreted the assembly actions that demonstrating the work’s effectiveness. The results are significant in that they will enable a robot to identify faults in the process in real-time and learn how to correct them.

The paper is organized as follows: in Sec. II, the simulation setup is introduced. In Sec. III the Pivot Approach is presented. In Sec. IV the snap verification system is described. In Sec. V simulation results are presented. In Sec. VI this work’s implications are discussed and in Sec. VII key findings are summarized.

II. SIMULATION SETUP

A 6 DoF Mitsubishi PA-10 manipulator was simulated in the OpenHRP environment [4]. A CAD derived female mold part was rigidly held by a two-fingered gripper mounted on the robot’s wrist, while a male mold part was rigidly fixed to the ground. The male part consisted of two cantilever snaps and a docking location as seen in Fig. 2. The two snap joints on the male part were simulated as spring joints. Angular data helped corroborate if the snap was successful. Snap assemblies were considered terminated when no further motion was possible. All signals were low-pass filtered using a digital recursive 1st order filter with a 0.05 cut-off frequency.

III. THE PIVOT APPROACH

As introduced earlier, to facilitate force signal interpretation into higher-level intuitive behaviors, one of the conditions is to implement a control strategy that generates similar sensory-motor signals across multiple trials of a task. This goal can be achieved first by using the same motion instruction set across trials but secondly and more importantly, attempting to constrain the motion after contact between the parts. The latter is more easily implemented in the snap assembly case. For most manufactured cantilever snap parts (cameras, phones, home appliances, etc.), their mechanical design is such that a “pivoting dock” is usually found and allows a female part to dock, constraint its motion, and rotate until a snap takes place. This is generally the case, for configurations of one, two, four or more cantilever snaps. These manufactured parts also contain internal hardware ridges that seek to facilitate the snap insertion once an optimal entry location has been achieved [4]. For this reason, the Pivot Approach was designed to exploit the inherent design in snap parts and yield a strategy that could be used to execute assemblies that would consistently yield similar patterns of force signature across trials. While the strategy is generalizable to configurations of growing complexity, the approach presented here dealt with snap parts that consisted of two cantilever snaps.

The Pivot Approach control strategy is described in detail in [4] and is visualized in 1. The latter begins with an optimal displacement from the homing position to the docking location’s neighboring region. From there a guarded approach follows until lateral contact is made with the pivoting dock and the female part is at angle from the horizontal. State 2 proceeds with a rotational motion towards the male part until contact is made with the snaps. This position is near optimal but requires alignment before the insertion can take place, State 3 displaces the wrist’s orientation to minimize moment errors. Upon alignment, state 4, initiates a compliant insertion that combines an insertion force while continuing to align the parts until the snap insertion takes places. At
this stage, state 5, applies a constant force to maintain the mating position between the parts. Each of these states is represented by colored boxes as seen in Fig. 4.

IV. SNAP VERIFICATION SYSTEM

The hierarchical taxonomy’s goal was to connect human proposals about actions like: “approaching”, “rotating”, “aligning”, “snapped”, and “mated” with LLB’s in a context-sensitive manner. One of the main challenges encountered in interpreting force signals is their inherent noise and spatio-temporal complexity. However, the force signals do inherently possess characteristics that describe the task at hand. The authors hypothesized that such characteristics could be extracted by looking at how temporal relative changes were associated to each other and contextualized by the state in which they occur. In so doing, intuitive behavior sequence’s can be extracted and their outcome examined. This level of discrimination is significant as it can be expanded to a real-time implementation and allow to reason about the state to perform corrective motions if necessary.

To bootstrap the approach, we partitioned the data into linear segments that approximate the data and classify the gradients according to magnitude per a small set of criteria. The next layer of abstraction examines at ordered-pair primitive sequences, and according to the gradient patterns presented and a small set of classification criteria, they are categorized into one of several types of motion compositions. The third layer abstracts sequences of motion compositions to identify LLB’s, while the fourth layer looks at what LLB’s are present in which states to determine if desired high-level behaviors are present. The final layer outputs the verification process results’ according to whether or not the desired sequence of HLBehaviors is present or not. A visualization of the hierarchical taxonomy can be seen in Fig. 3.

![Hierarchical Taxonomy for Cantilever-Snap Assembly Verification](image)

Fig. 3. Hierarchical Taxonomy for Cantilever-Snap Assembly Verification.

A. PRIMITIVE LAYER

The primitives layer requires that each signal is partitioned into linear segments of data that closely approximate the original signal. Linear regression in concert with a correlation measure (the determination coefficient $R^2$) is used to partition the data whenever a minimum correlation threshold is crossed. If the determination coefficient drops under a given threshold the linear fit is partitioned and a new regression is started. The $R^2$ coefficient is a correlation measure that studies the ratio of the squares of the residual errors between the original data $y$ and the fit data $\hat{y}$ to the sum of the variance $\sigma_y^2$ as shown in Eqtn. 1.

$$R^2 = 1 - \frac{\sum (y - \hat{y})^2}{\sigma_y^2}$$

The threshold used to partition the data was set at 0.70, such that if the correlation dropped to under 70%, a linear segment or “partition” would be generated, and a new one would start at the next data point. The data was traversed by a window equal to five data points (the data was sampled at a frequency of 1kHz by the simulation). The threshold values and the window length were empirically selected to partition the data sufficiently to capture relevant changes in the signals.

Each partition was accompanied by a data structure with seven types of information about itself: the average value across data points, the maximum value, the minimum value, the start time, the end time, the gradient value, and a gradient label. With respect to the latter, nine gradient labels (positive impulse, ‘pimp’; big, medium, and small positive gradients, ‘b/m/spos’; constant gradients, ‘const’; and their negative equivalents, ‘nimp’, ‘b/m/s/neg’) were assigned according to ranges summarized in Table I. The classification first attempts to separate instances of data in which contact or mating takes place. On the one hand, contact phenomena is characterized by very rapid and large changes in force signals, almost approximating an impulse. To this end, positive and negative impulses were categorized for gradients with values greater or less than 1000. On the other hand, for mating situations, there is little or no change in force, for this reason a constant label was assigned to signals with gradient values less than the absolute value of 2. In between these two extremes we chose to have three gradient categories for both positive and negative signals to give a general idea of the magnitude change registered for a signal. Fig.4 shows how the segmentation looks like across all five states (which are represented by five colored boxes) for the force signal in the x direction.

<table>
<thead>
<tr>
<th>Gradient Values Classification for the Primitives Layer.</th>
</tr>
</thead>
<tbody>
<tr>
<td>pimp</td>
</tr>
<tr>
<td>bpos</td>
</tr>
<tr>
<td>mpos</td>
</tr>
<tr>
<td>spos</td>
</tr>
<tr>
<td>const</td>
</tr>
<tr>
<td>nimp</td>
</tr>
<tr>
<td>bneg</td>
</tr>
<tr>
<td>mneg</td>
</tr>
<tr>
<td>sneg</td>
</tr>
</tbody>
</table>
B. Composites

The next layer of abstraction identifies seven basic motions compositions (MC) by looking at ordered-pair sequences of primitives. The MC layer set is comprised of: adjustment, increase, decrease, constant, contact, positive contact, negative contact, and unstable motions. The positive and negative contacts imply the sign of the (gradient) of the action.

A protocol was followed to minimize the effects of noise or erroneous segmentation. With respect to adjustments, primitives with big-small positive or negative gradients were considered as a positive or negative primitive category respectively. If a positive grouped primitive was followed by a negative grouped primitive an adjustment classification would be assigned to the ordered pair. Adjustments are motions in which the wrist records a quick “back-and-forth” motion typically seen during alignment or insertion operations as the force controller tries to minimize residual errors. The reason to group positive and negative gradients is to maximize the likelihood of group adjustments even when the rate of change may be slightly different. Furthermore, for this particular category, we used a window of two data points instead of one to look for a matching pair (all other categories looked at the contiguous primitive). That is, if after finding a positive or negative gradient, and if the next data point was not negative or positive respectively, we would look at the next data point to look for a pair. Such procedure mitigates the presence of spurious signals that could prevent the proper grouping of an adjustment movement.

The ordered pair groupings for motion composition classification are summarized in Table II. Note that the table contains sub-tables representing five primitive groupings. The first primitive is in bold text followed by a listing of second primitives and the corresponding motion composition and label used in the plot as illustrated in Fig. 5. As with the primitives layer, 11 pieces of information were collected for each MC: composition label, average value, root means square value, amplitude, the labels of the first and second primitives, the starting and ending times for both primitives, and the average time for both primitives.

1) Refinement: After the MCs are generated, a refinement phase was used to filter less significant signals and augment more significant signals. To do so, the compositions were analyzed under three contexts: (1) a composition’s time duration, (2) a composition’s amplitude magnitude, and (3) composition repetition patterns.

- Time Duration Context: this filter examines two contiguous MCs. If either composition is seven times bigger than the other, the smaller composition is merged to the larger one and all data is updated correspondingly. The duration ratio was determined empirically.

- Amplitude Value Context: this filter pertains to the formation of adjustment signals and constant signals. We considered three possibilities: (i) If there are contiguous primitives of types PC/NC or NC/PC, and if their amplitude is ten times smaller than the largest amplitude registered in the assembly, then treat them as an adjustment. This criteria seeks to disambiguate real contact signals and false ones by looking at their amplitude. Real contacts are characterized by large values. (ii) Similarly, if their is either an increase followed by a decrease and vice-versa, and both compositions have a similar amplitude (within 50% of each other and they have a similar average value (100%) of each other, then merge as an adjustment and update the data correspondingly. (iii) If there is a sequence of an increase followed by a constant, or a decrease followed by a constant and vice-versa, and they have a similar amplitude (150%)and similar average value (100%), then merge them as a constant and update their information. This last filter targets small noisy signals that appear as increases or decreases but that in effect are constants. The amplitude threshold value is larger here to give more possibilities of catching increases or decreases within the narrow range of the constant’s amplitude. - Repeated Compositions: the last filter takes signals that repeat and merges them as one. This filter is run iteratively until no more repetitions occur in the data.

The post-refinement composition layer results are shown in Fig. 5 for a force signal sample in a Pivot Approach trial.

C. Low-Level Behaviors

The taxonomy’s third layer considers motion composition ordered pairs along with signal duration and amplitude to yield classifications. Eight LLB classifications were derived and labeled as: push, 'PS', pull, 'PS', contact, 'CT', fixed,
TABLE II

<table>
<thead>
<tr>
<th>Combination</th>
<th>Category</th>
<th>Label</th>
<th>Combination</th>
<th>Category</th>
<th>Label</th>
<th>Combination</th>
<th>Category</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>Pimp</td>
<td>a</td>
<td>Positive</td>
<td>pos contact</td>
<td>pc</td>
<td>Positive</td>
<td>increase</td>
<td>i</td>
</tr>
<tr>
<td>Negative</td>
<td>adjustment</td>
<td>a</td>
<td>Negative</td>
<td>pos contact</td>
<td>pc</td>
<td>Negative</td>
<td>decrease</td>
<td>d</td>
</tr>
<tr>
<td>Positive</td>
<td>increase</td>
<td>i</td>
<td>Constant</td>
<td>pos contact</td>
<td>pc</td>
<td>Constant</td>
<td>constant</td>
<td>k</td>
</tr>
<tr>
<td>Pimp</td>
<td>pos contact</td>
<td>pc</td>
<td>Pimp</td>
<td>unstable</td>
<td>u</td>
<td>Pimp</td>
<td>pos contact</td>
<td>pc</td>
</tr>
<tr>
<td>Nimp</td>
<td>neg contact</td>
<td>nc</td>
<td>Nimp</td>
<td>contact</td>
<td>c</td>
<td>Nimp</td>
<td>neg contact</td>
<td>nc</td>
</tr>
<tr>
<td>Negative</td>
<td>Nimp</td>
<td>a</td>
<td>Positive</td>
<td>neg contact</td>
<td>nc</td>
<td>Negative:</td>
<td>neg contact</td>
<td>nc</td>
</tr>
<tr>
<td>Constant</td>
<td>decrease</td>
<td>d</td>
<td>Constant</td>
<td>neg contact</td>
<td>nc</td>
<td>Constant</td>
<td>neg contact</td>
<td>nc</td>
</tr>
<tr>
<td>Pimp</td>
<td>pos contact</td>
<td>pc</td>
<td>Pimp</td>
<td>contact</td>
<td>c</td>
<td>Nimp</td>
<td>unstable</td>
<td>u</td>
</tr>
<tr>
<td>Nimp</td>
<td>neg contact</td>
<td>nc</td>
<td>Nimp</td>
<td>unstable</td>
<td>u</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

'FX', alignment, 'ALIGN', shift, 'SH', and noise, 'N'. The LLB formulation criteria is similar to those at the MC level. That is, for a pair of increase MCs labels, or decrease MCs labels, or constant MCs labels or adjust MCs labels; pull, push, fixed, or adjust LLBs are assigned respectively. As for contacts, if there is a positive contact followed by a negative one, or vice-versa, a contact LLB is assigned. One major difference between the MC level and the LLB level is introduction a shifting behavior ‘SH’. Shifts and alignments are similar but differ in that, whenever there are two contiguous adjustment compositions, if the second composite’s amplitude is larger than the first, label it as ‘SH’ LLB, if smaller label it ‘ALGN’.

With regards to the time duration context, if any motion composition lasts more that 100 milliseconds, it can by itself be a low-level behavior, or if the contiguous composition is of the same classification they can also merge correspondingly. If any composition is less than the allotted duration and it does not have a matching pair, it is considered a noisy signal. With regards to the amplitude context, if there are two adjustments within a window of 2 data points, and their amplitudes decrease, render such a pair as an alignment, otherwise consider it a shift (or growing de-alignment). As for paired increase, decreases, and constants, they will yield pull, push, and fixed low-level behaviors correspondingly. Finally, as for contacts, if there is a positive contact followed by a negative one, or vice-versa, or even a stand-alone contact motion primitive, render this is a low-level contact behavior.

1) Refinement: The LLB layer is also followed by a refinement phase. The latter filters based on the same three contexts as used before:

- Time Context: this filter examines two contiguous behaviors (except for contacts). If either behavior is five times bigger than the other, then merge towards the longer behavior and update the data correspondingly. LLB’s are longer than compositions, so this threshold value is to be smaller than the one used for the composition’s time duration filtering.

- Amplitude Context: the amplitude context pertains to alignments and shifts and there are four possible scenarios: (i) If there is a push-pull pair in either order and they have similar amplitudes (150%) and similar average values (100%) render then an alignment. (ii) If there is a shift followed by an alignment, or an alignment followed by a shift, where the second behavior has a smaller amplitude, then merge these as an alignment. This kind of merging is interesting because it can only be seen at this level of abstraction. While there may be a contiguous alignment-shift pair that was irreconcilable earlier, it can now be identified as an alignment. The same is done for a shift. (iii) Finally if there is an alignment followed by a pull or push or vice-versa and they have similar amplitudes (50%) and similar average value (100%), then merge as an alignment. In this case, after the previous refinement steps have been executed, if there are outstanding alignment-push or -pull pairs, the second behavior is a considered a continuation of the alignment and is merged. Shifts are treated similarly.

- Repeated Behaviors: As in the compositions layer, any two repeated behaviors can be merged as one. The post-refinement LLB layer is shown in Fig. 6 for a sample signal in the Pivot Approach.

Fig. 6. LLB Layer: behaviors in this layer are generated from a sequence of motion composites. Their labels are shown below the signal.

D. High-Level Behaviors

The fifth layer contextualizes the process monitoring by asking what low-level behaviors principally describe the high-level human apropos behaviors found in the Pivot Approach: Approach, Rotation, Alignment, Snap Insertion,
and Mating¹.

Then, if key combinations of LLB’s across the six force axes for a specific state are identified, then a certain HLB can be ascertained. For each state and corresponding HLB an LLB or sequence of LLB’s are matched with a particular force axis as part of the criteria: The reasoning behind the selection of LLB’s and axis for the Pivot Approach is simple and intuitive. For state 2, the rotation state, the wrist maintains a constant force along the z-direction, while the force along the y-direction diminishes as the wrist aligns itself with male part. The rotation about the x-axis can be seen through a series of large alignments along the moment’s x-direction. For state 3, all axes are aligning in some form. For force elements, there is an alignment in position, for moment elements there is an alignment in orientation. The only exception to this is the moment about the z-direction. A pattern emerges where the moment axis that corresponds to the wrist’s direction of motion for the insertion (i.e. the z-axis for the Pivot Approach) experiences little to no change throughout the assembly due to the nature of parts in the assembly task. For state 4, in the insertion state, Rusli studied typical force patterns for manually effected snap assemblies and states that initial resistance is characterized the insertion until the snap-catch slips behind the undercut in the mating part, at which time an interlock occurs. In other words, one a large increase in force is expected upon contact, followed by a large decrease in force. Hence, we expect to see a contact label followed by an alignment label. Other axis can expect to experience an alignment at this stage. Finally, for the mating state, all signals should present no motion change and thus be classified with a FX behavior.

The fourth layer results are shown in Fig. 7. If the high-level behaviors can be ascertained, they print on the plot in green color. If they cannot be verified, they plot in red color representing failure.

### E. Verification Layer

The fifth and last layer simply declares whether the assembly was entirely successful or not.

¹In actuality, we do not directly assess the Approach stage given that there are no contact forces at this stage. But if the rotation analysis is successful we assume that the approach was too.

---

**V. RESULTS**

In this section we present the result of a trial in which each of the six force signals is analyzed and combined to do a system verification. The results are visualized in Fig. 8. The visualization of all the results in Fig. 8 contains very intuitive patterns between the LLB’s and the HLB’s for the Pivot Approach. As noted in Sec. IV-D, the rotation behavior is clearly identified by the pull in the $F_y$ direction, the fixed position in the $F_z$ direction, and by the alignment motion in the $M_x$ direction. The align behavior is characterized by alignment LLB’s across all states. The snap insertion is distinguished by the high-force contact behavior characteristic along the direction of insertion $F_z$. The last state is as expected characterized by fixed behaviors as at the mating stage no further motion should be experienced.

The system also predicted cases in which the assembly was a failure. One example is shown in Fig. 9. During this attempt, the force controller encountered difficulties in properly aligning the parts as can be seen in the shifts in $M_x$ and $M_y$. The failure to align not only result in a lack of contact behavior in $F_z$ but also a motion away from mating as visualized in the fifth state for $M_x$ and $M_y$, and as well as by the fact that the first snap did not converge to its home position. The verification system noted this by displaying the “Snap” and “Mating” states in red and by the “Failure” label at the top of each plot. The reason for failure may be due to unexpected force values that sometimes are rendered by our simulation program upon contact. OpenHRP was originally designed for walking humanoid robots and its dynamic engine is not well suited for small contacts. Our system was run on a set of six simulation trials with 100% accuracy rate in its prediction.

**VI. DISCUSSION**

This work effectively demonstrated that by limiting the way a cantilever snap assembly can be generated, namely the Pivot Approach, and that by using a small set of classification categories that encode relative change at different abstraction levels, force signals can be interpreted at a human apropos level of intuition that correspond to the assembly’s action states. The method effectively determined the status of successful and failed assemblies.
This work offers a simple approach to analyze behaviors at each stage of the cantilever snap assembly. We propose that the snap assembly needs to be classified according to class and that the approach needs to be consistent to yield similar patterns of force signatures across the tasks. Our current work offers a viable approach to understand and verify the cantilever-snap assembly process, and it can be extended to increasingly complex geometries of cantilever snaps. It can also be extended to service robots that need to interact with objects where snap fasteners are also common.

The classification labels devised in layers 1-3, were obtained by listing basic blocks in which signals can change. The first level considers relative magnitudes while the second and third layers abstract changes to actions to simple behaviors. Yet the categorizations across levels are similar and connected. That is to say consecutive positive gradients are connected to an increase which is connected to a pull. Similarly for other categorizations. It may be possible that this methodology could be extended to other assembly operations like the peg-in-hole case, perhaps because the geometries are somewhat similar. However, the authors think it unlikely, that this method as it stands, would work for annular or torsional snaps in which the snap parts are circular or spherical and include rotational motions.

A review of the literature, reveal that the patterns of force-moment signals produced in our experiment agree with those described in the works of Rusli et al. [8] where force signals were analyzed upon the manual assembly of cantilever snap fits; our results also bear resemblance with those obtained by Stolt et al. in the development of a force controller for snap assembly [5].

Some limitations in our work lie in the selection of parameter values. From the determination coefficient threshold in the linear fit, to the gradient margin values, to the time duration, amplitude value, and average value margins used in the refinement phases; changes in these values could alter the verification result. An important challenge for the future is to study if there is a method that could extract these properties autonomously. We will soon start on an offline optimization procedure that determines optimal parameters for system verification in future assemblies. Furthermore, the authors also believe it will be important to more relevantly include other contextual data such as the end-effector position. This remains part of our future work.

The results obtained so far open interesting new directions in real-time process verification, fault recovery, learning and optimization. The current work will be migrated to a humanoid robot performing the pivot approach on a real
manufactured camera part. The online process would require the system to predict the assembly’s success likelihood for each state and not defer until completion to determine the task’s result. Thus, the verification system would adopt a probabilistic nature. Furthermore, upon likely failure, information from the LLB layer and the motion composition layers, could be used to determine what corrective motion to take. Moreover, the degree with which the robot should perform the corrective motion could be extracted from the value and gradient properties captures through the system’s compositions. Finally, learning mechanisms could then be implemented on top of the systems prediction and response mechanisms to learn typical failure and appropriate corrective measure scenarios.

VII. CONCLUSION

This work proposed a process monitoring and system verification method for cantilever-snap assemblies. This work effectively demonstrated that by limiting the way a cantilever snap assembly can be generated, namely the Pivot Approach, and that by using a small set of classification categories that encode relative change at different abstraction levels, force signals can be interpreted at a human apropos level of intuition that correspond to the assembly’s action states. The approach will be expanded to perform probabilistic online process monitoring and allow a robot to reason about the assembly’s state and take corrective or preventive actions to ensure the success of the task.

REFERENCES