ARTICLE INFO
Article ID: 05-11-01-0001

doi:10.4271/05-11-01-0001

# Classification of Contact Forces in Human-Robot Collaborative Manufacturing Environments

**Ran Zhao,** University of Nottingham

***Adrien Drouot and Svetan Ratchev,*** *University of Nottingham*

## Abstract

This paper presents a machine learning application of the force/torque sensor in a human-robot collaborative manufacturing scenario. The purpose is to simplify the programming for physical interactions between the human operators and industrial robots in a hybrid manufacturing cell which combines several robotic applications, such as parts manipulation, assembly, sealing and painting, etc. A multiclass classifier using Light Gradient Boosting Machine (LightGBM) is first introduced in a robotic application for discriminating five different contact states w.r.t. the force/torque data. A systematic approach to train machine-learning based classifiers is presented, thus opens a door for enabling LightGBM with robotic data process. The total task time is reduced largely because force transitions can be detected on-the-fly. Experiments on an ABB force sensor and an industrial robot demonstrate the feasibility of the proposed method.

### History

| | |
|---|---|
| Received: | 07 Aug 2017 |
| Revised: | 09 Nov 2017 |
| Accepted: | 14 Nov 2017 |
| e-Available: | 02 Apr 2018 |

### Keywords

Human-robot collaborative manufacturing, Contact force classification, Machine learning

### Citation

Zhao, R., Drouot, A., and Ratchev, S., "Classification of Contact Forces in Human-Robot Collaborative Manufacturing Environments," *SAE Int. J. Mater. Manuf.* 11(1):2018, doi:10.4271/05-11-01-0001.

ISSN: 1946-3979
e-ISSN: 1946-3987



# Introduction

Sensor technologies have been employed for robotic applications across a large variety of fields in industrial scenarios. Among them, force/torque sensors play an important role because industrial robots are usually position-controlled, employed for performing tasks quickly, accurately, and repeatedly because they do trajectory following very well [1, 2]. But this control strategy is not suitable for tasks that require interaction with the environment or the human. In the former case, force sensors are applied to reduce uncertainties in position [3] whereas in the latter case, physical human-robot interaction (pHRI) can be achieved by introducing an additional force sensor [4].

In modern industry, robots are increasingly expected to perform in hybrid manufacturing cells, where humans and robots can collaborate to some extent to accomplish tasks that human need to participate. In order for enhanced flexibility and high productivity, robot controllers need to be equipped with various sensors. In the case of human-robot collaborative manufacturing, force sensors are essential to expand the robot capabilities.

Machine learning applications of force/torque sensors have previously been used in industrial scenarios. [5] used a Support Vector Machine (SVM) and a single-axis force sensor to distinguish successful and failed assemblies. SVM was also introduced in [6, 7] to detect tool breakage in milling processes. Different approaches were presented to monitor assembly tasks in order for error detection using Hidden Markov Model [8] and hierarchical taxonomy [9]. Stolt *et al.* [10] applied several classifiers based on machine learning algorithms, such as Least-squares and Boosting to detect force/torque transients in robotic assembly tasks.

This article proposes a new application of force/torque sensors in combination with machine learning techniques for industrial robots.

The purpose is to automatically classify different interaction forces applied on the robot in a human-robot collaborative manufacturing environment. Figure 1 illustrates the overall procedure. Sensor data is firstly collected and then input to an on-line multiclass classifier together with training samples. The output is a force class that is allocated to trigger a corresponding robot action. The advantage is that the robot programming is largely simplified by the physical interaction. The robots can be controlled in an intuitive way. This paper only considers contacts between the human and the robot.

**FIGURE 1** The overall procedure of the proposed force classification scheme in robotic applications.

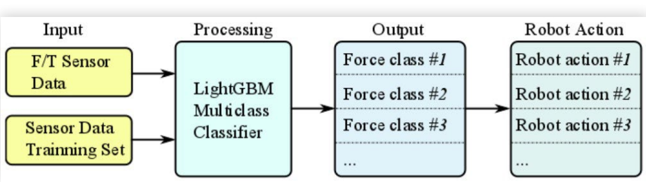
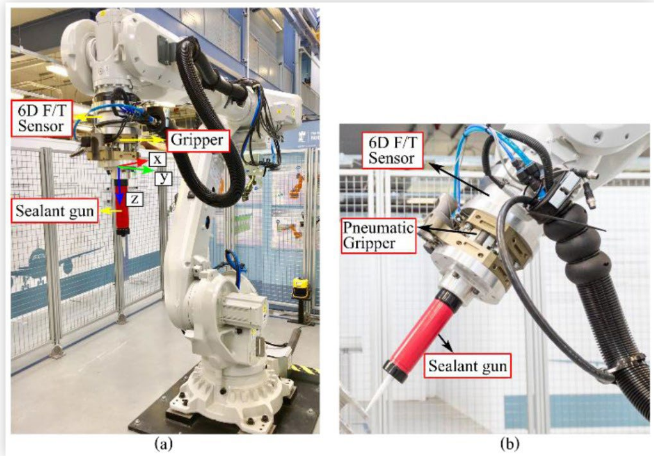





**FIGURE 2** (a) The setup of sealant application in a robotic cell and the coordinate frame used in the task. (b) The end-effector details: the sealant gun, the pneumatic gripper, and the force/torque sensor.





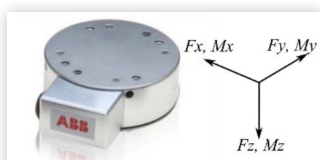

# Scenario Design

## Robot System and Sensor

This paper selects a collaborative robot task in a force-controlled sealant application. The robot holds a sealant gun while the human can drive the end-effector to the desired position that is to be sealed. Figure 2(a) illustrates the setup of the sealant application.

The robot used in this task is the ABB industrial manipulator with six joints. The robot is controlled via the IRC5 controller. A force control system is integrated into this controller, which allows for the modification of the reference for the low-level joint control loop. An ABB 6-DOF force/torque sensor is mounted at the wrist between the last joint and the pneumatic gripper (as shown in Figure 2(b)). The capacity of this sensor is also shown in Figure 3.

**FIGURE 3** The force/torque sensor used in the cell and its specification [11].




Sensor2500 Specification

| | Capacity | Overload Capacity |
|---|---|---|
| *Fx* | 2500 *N* | 25000 *N* |
| *Fy* | 2500 *N* | 25000 *N* |
| *Fz* | 6250 *N* | 62500 *N* |
| *Mx* | 400 *Nm* | 400 *Nm* |
| *My* | 400 *Nm* | 400 *Nm* |
| *Mz* | 400 *Nm* | 400 *Nm* |

## Design of Force Classes and Robot Actions

The sequences of the sealant task are as follows:

$S_1$: human attaches the sealant gun to the robot
$S_2$: human drives the end-effector to start position
$S_3$: robot does the force controlled sealant task
$S_4$: human detaches the sealant gun

This paper then chooses five contact force classes with respect to the designed sequences in the collaborative task. Table 1 indicates the proposed contact force types and the corresponding robot actions. In $S_1$, after human attaching the sealant gun to the robot, a clockwise torque along $Z$ direction will command the robot close the gripper to lock the tool. Similarly, gripper will be opened with a counter-clockwise $M_z$. Continuous force in any direction will initiate the admittance control mode, which enables the human drive the robot to desired position manually. A short (about 1 s) press on the positive $Z$ direction ($F_z$) is used to start the robot sealant program. Any unexpected random contacts will be considered as interferences or collisions in all of the sequences.

# Procedure

This section describes the systematic procedure to get a multi-class classifier for different force events. First of all, data need to be collected and processed. Then the classifier has to be trained using these data. For simplicity, only the continuous force/torque class described in this section. The procedure of other force classes is the same.

## Data Acquisition and Pretreatment

The force/torque sensor contains six channels. In regard to continuous force $\mathcal{F}\_confm$ or unexpected contact $\mathcal{F}\_unept$, it might be applied on any direction. In order for an appropriate training sample, ten recordings were made on each direction. The total number of the recordings was thus 60. For other classes that only have force applied on one axis, the number of recording is 5. Figure 4 illustrates a data example of $\mathcal{F}\_confm$, $\mathcal{F}\_clomz$, and $\mathcal{F}\_unept$, respectively.

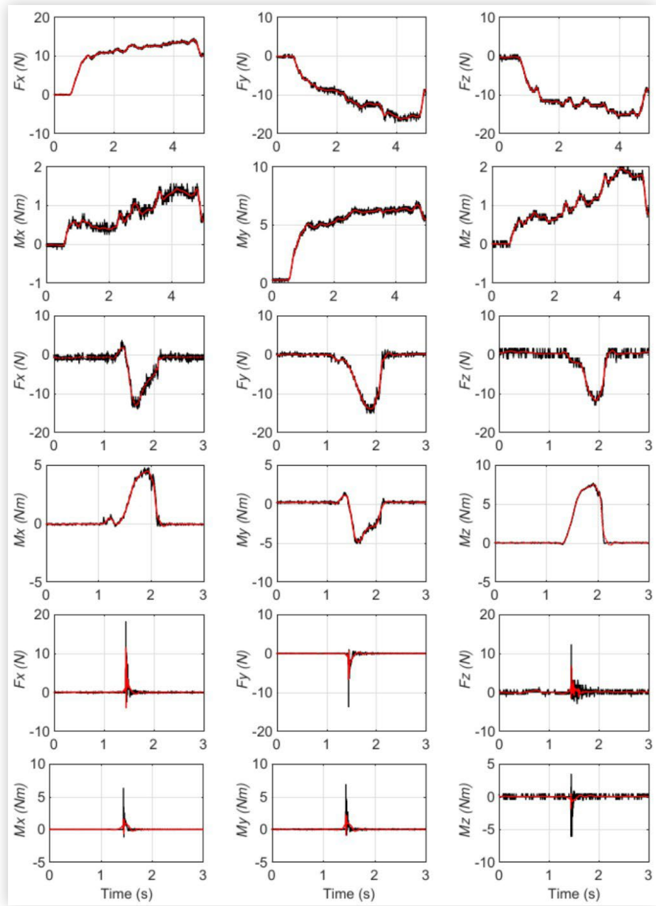
**TABLE 1** Force class and the corresponding robot action in all sequence.

| Sequence | Force class | Description | Robot action |
|---|---|---|---|
| $S_1$ | $\mathcal{F}\_clomz$ | Clockwise $M_z$ | Close gripper |
| $S_2$ | $\mathcal{F}\_confm$ | Continuous $F/M$ | Admittance control |
| $S_3$ | $\mathcal{F}\_fz$ | Short press in $z$ direction | Execute sealant task |
| $S_4$ | $\mathcal{F}\_cclmz$ | Counter clockwise $M_z$ | Open gripper |
| *All* | $\mathcal{F}\_unept$ | Unexpected contacts | Do nothing |





**FIGURE 4** An example of the recorded data from forces applied on the robot end-effector. The force/torque directions refer to the feature coordinates defined in Figure 2(a). The black curves are the original data while the red curves are the de-noised data. The first six figures: a continuous force; the middle six figures: clockwise torque; the last six figures: a random contact.






The force classes might seem to be easily recognized in these data, but the classifier must have the ability to do it with only a subset of the data. For the real-time requirement, the smaller subset is preferable in the classification process. The data were pre-scaled such that all force/torque components got approximately the same magnitude. This should also make the problem of training the classifiers better numerically conditioned.

## Data for Training

There are an interesting force event and a lot of background data in each recording. It is not desirable to use all of the recorded data for training. A choice is to select a sample before the force event and a sample after that, then put after one another with the force event in between.

Considering the cases in the sealant task, this paper chose 100 + 100 + 100 data as a training sample for each channel. That is, 100 data before the force, 100 data after the force was applied (detected by predefined threshold), then the subsequent 100 data were selected.

## Cost Function

A cost function is used in order to optimize the classification performance. In the scenario considered in this paper, misclassifying a force event is worse than missing a force event in the data set, because the robot will act in a totally different way. This may cause danger in human-robot collaboration. So a misclassification is a false negative whereas a missing classification is a false positive. As computation time is another significant factor for real-time applications, this paper proposed the cost function as follows:

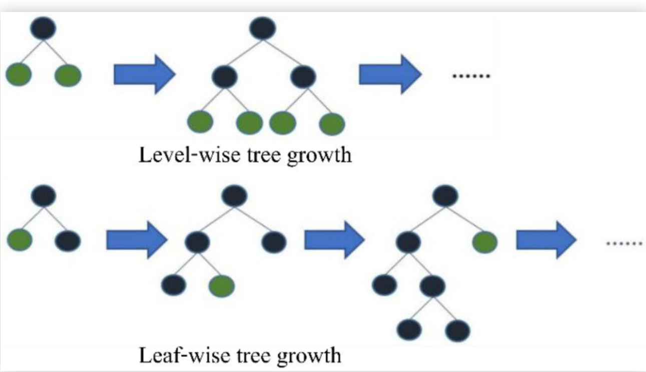
$$C = cost_{FP} * n_{FP} + cost_{FN} * n_{FN} + cost_{CT} * t \qquad \text{Eq. (1)}$$

$cost_{FP}$, $cost_{FN}$ and $cost_{CT}$ are the costs of false positives, false negatives, and computation time, respectively. $n_{FP}$ and $n_{FN}$ are the numbers of false positives and false negatives while $t$ is the time (in $ms$) used during the classification.

## Classifier Training

**LightGBM** LightGBM is a gradient boosting framework based on decision algorithms, used for many machine learning task, such as ranking and classification, etc. LightGBM uses the histogram based algorithms, which bucketing continuous feature values into discrete bins, to speed up training procedure and reduce memory usage [12]. It produces much more complex trees by following leaf wise split approach rather than a level-wise approach which is the main factor in achieving higher accuracy. It is capable of performing equally good with large datasets with a significant reduction in training time. LightGBM grows a tree by leaf-wise. It will choose the leaf with max delta loss to grow. When growing the same number of leaf, leaf-wise can reduce more loss than a level-wise algorithm, as shown in Figure 5. The details of the leaf-wise algorithm can be found in [13].

**FIGURE 5** The difference between the level-wise algorithm and the leaf-wise algorithm.






Two more choices of classifiers were used for comparison, namely Support Vector Machine (SVM) and Least-squares (LS). These classifiers were used because they were both quite computationally cheap and that they had no extra parameters to tune.

**Training a Subset of Data** The recorded data contained six channels of force/torque information. One choice for the training set is to use a combination of these channels. The number of channels used is denoted as $n_{ch}$. It has a positive correlation with the classification result whereas a negative correlation with the computational complexity (e.g. computation time). The best choice of $n_{ch}$ can be obtained by minimizing the cost function (1).

An alternative way is to simplify the problem by merging the force channels and the torque channels receptively. That is, each recording is divided into two subsets: $sub_F$ and $sub_M$, where

$$sub_F = Fx + Fy + Fz \qquad \text{Eq. (2)}$$

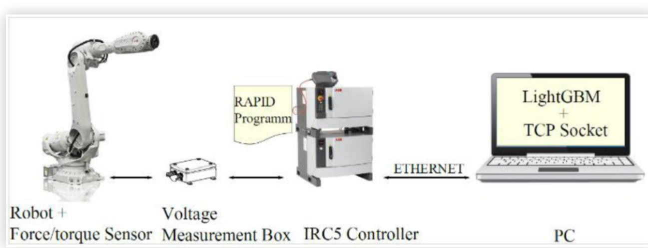
$$sub_M = Mx + My + Mz \qquad \text{Eq. (3)}$$

Both of the two approaches were evaluated in the experiment section.

## Implementation

The training procedure described in this section was implemented in C++ on an Intel® Core (TM) i7 CPU 2.80 GHz machine. Software running on the robot was written by the ABB RAPID programming language. It is composed of two independent tasks: a publisher and a subscriber. The publisher continuously monitors the robot's force/torque data. The subscriber receives commands from the PC program. TCP socket was used for communication between PC and the robot, such as sending the classification results. TCP sockets are selected for network connection and data transmission in order to transport the data in a correct sequence. Although UDP sockets allow faster communication, there is no guarantee that the packets sent would reach at all. The force/torque sensor was used with a configured low-pass filter and it was sampled at 100 Hz. The system architecture of the implementation is given in Figure 6.

**FIGURE 6** The system architecture of the implementation.



# Experimental Results

The experiment was conducted for the continuous force/torque, that is, the force class $F_{confm}$. Figure 7 shows the classification results when all different combinations of channels were tested. For example, 6 experiments were conducted for the case of single channel while 15 experiments for the two-channel case. The marks on the figures shows the number of the false positives and false negatives with different number of channels used. Figure 7 indicates that, with the increase of the number of channels, $n_{FN}$ decreased for all of the three approaches. That means the chance of misclassification reduced largely due to more captured sensor data. Among these three approaches, LightGBM had an overwhelming performance when four or more than four channels were used. While the performance was not satisfactory when sensor data was collected from fewer channels. It is because LightGBM has advantages in large dataset processing. The last sub-figure in Figure 7 shows the average consumed time for all approaches with different channel numbers. The consumed time extended as the channel number increased because more data required more time for processing.

Table 2 illustrates the average total costs associated with the experiment. The costs of false positives and false negatives were defined as $cost_{FP} = cost_{FN} = 1$ while the cost

**FIGURE 7** Classification result of three different classifiers. All different combinations of channels were tested.

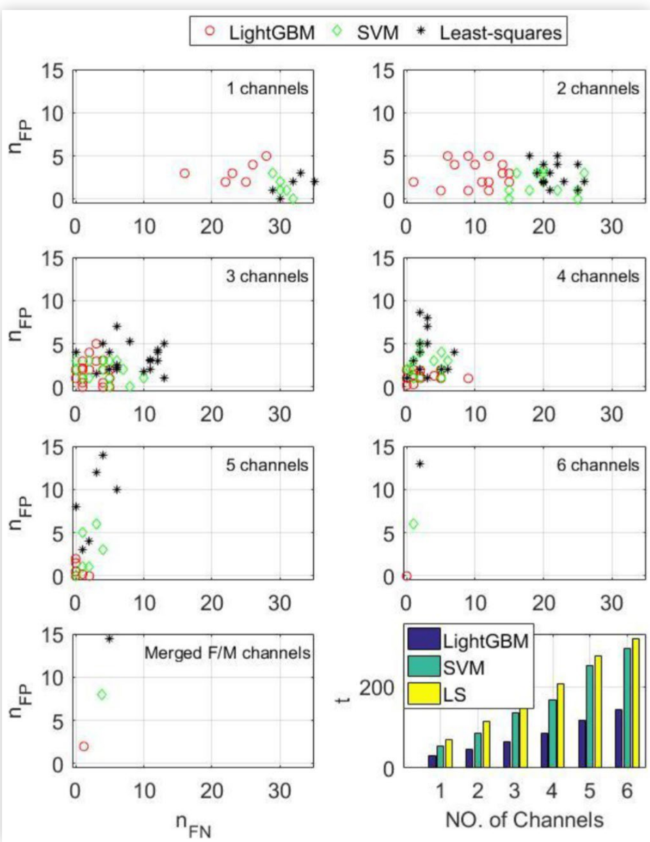







**TABLE 2** Result from training: the average cost $C$ of the three classifiers for the force class $F_{confm}$ $(cost_{FP} = cost_{FN} = 1,\ cost_{CT} = 0.2)$.

| | LightGBM | SVM | Least-squares |
|---|---|---|---|
| 1 channel | 33.167 | 42.833 | 48.333 |
| 2 channels | 22.067 | 38.44 | 47.4 |
| 3 channels | 18.507 | 39.167 | 48.106 |
| 4 channels | 17.115 | 33.25 | 41.655 |
| 5 channels | 24.8 | 54.7 | 66.167 |
| 6 channels | 28.6 | 65.8 | 78.6 |
| Merged channels | 30.06 | 59.934 | 72.76 |



of time consumed was set to $cost_{CT} = 0.2$ as time was less important as the accuracy in this experiment. The range of the *cost* is [0, 1]. This is a user-defined value so it is able to be changed with respect to the experiment setup. The goal of the approach is to minimize total cost, thus the minimum total cost indicates the best results in the form of accuracy and used time. It can be clearly seen that LightGBM had a better performance in the force classification than the SVM and LS. Force classification with a combination of four channels gave the minimum total cost which was only 17.115 in all of the three classifiers. It is as expected because more channels will lead to more accurate results but will also cause larger computation load thus longer time used. Merging the force and torque channels shortened the training time but the accuracy did not get better. The reason is that in some cases, force/torque values may counteract each other when merging them. Therefore the total cost of merged channels is even larger than the case of two channels. Same results were obtained from more experiments that have been conducted for other force classes.

# Discussion

Although LightGBM has been released for a period of time, it is still new to the robotics community. This paper hereby described a robotic application of LightGBM in an industrial setup, thus opened a door for enabling it with robotic data processing. As Xgboost platform is more commonly used in robotics, we have also conducted the experiment with the Xgboost algorithm and the results show that LightGBM is no better than the Xgboost in this setup. That is because LightGBM uses Leaf-wise tree splitting which enables it to converge much faster. It is more efficient in dealing with large datasets. In our experiments, when we trained the sensor data of a single or two channels, Xgboost has similar or faster speed and the accuracy is also similar (sometimes Xgboost is better). When 6 channels of data are used and more recordings (such as 50 recordings for all channels) are taken, LightGBM is slightly better in the accuracy but much better (about 2 times faster) in the training speed. In addition, when dealing with small datasets, some parameters such as the maximum depth of each tree should be tuned to avoid overfitting.

# Summary/Conclusions

This paper described an industrial application using force/torque sensor. A systematic procedure was presented for classification of interactive contact states between humans and robots. A number of force classes in a robotic sealant task were proposed to initiate different robot actions. The LightGBM platform which is currently very new and less documented was first applied to a robotic application. Experimental results verified the effectiveness of this new platform in robotic data processing. By using the force classifier, the robot programming for pHRI was largely simplified. The total task time was averagely reduced by 14.6% because transitions among steps were detected on-the-fly, rather than manually. This paper will open a door for various robotic applications using the LightGBM platform.

## Acknowledgments

We gratefully acknowledge funding from Innovate UK No. 38279-263183 (VIEWS).

## Definitions/Abbreviations

**LS** - Least-squares

$n_{FP}$ - Number of the false positives

$n_{FN}$ - Number of the false negatives

**OvR** - One-vs.-rest

**SVM** - Support Vector Machine

**Sub**$_F$ - The subset of the force data

**Sub**$_M$ - The subset of the torque data

# References

1. Shi, J., Jimmerson, G., Pearson, T., and Menassa, R., "Levels of Human and Robot Collaboration for Automotive Manufacturing," in *Proceedings of the Workshop on Performance Metrics for Intelligent Systems*, Mar. 20-22, 2012, College Park, MD, 95-100. ACM, 2012.
2. Heyer, C., "Human-Robot Interaction and Future Industrial Robotics Applications," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 18–22, 2010, Taipei, Taiwan, 4749-4754. IEEE, 2010.
3. Hasan, M., "A Robust Real-Time Position and Force (Hybrid) Control of a Robot Manipulator in Presence of Uncertainties," in *Proceedings of IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, MFI2003*, Aug. 1, 2003, Tokyo, Japan, 138-143. IEEE, 2003.
4. Reed, K.B. and Peshkin, M.A., "Physical Collaboration of Human-Human and Human-Robot Teams," *IEEE Transactions on Haptics* 1(2):108-120, 2008. doi:10.1109/TOH.2008.13.
5. Rodriguez, A., Bourne, D., Mason, M., Gregory, F. et al., "Failure Detection in Assembly: Force Signature Analysis," in *2010 IEEE Conference on Automation Science and Engineering (CASE)*, Aug. 21-24, 2010, Toronto, ON, Canada, 210-215. IEEE, 2010.
6. Cho, S., Asfour, S., Onar, A., and Kaundinya, N., "Tool Breakage Detection Using Support Vector Machine Learning in a Milling Process," *International Journal of Machine Tools and Manufacture* 45(3):241-249, 2005. doi:10.1016/j.ijmachtools.2004.08.016.
7. Hsueh, Y.-W. and Yang, C.-Y., "Prediction of Tool Breakage in Face Milling Using Support Vector Machine," *The International Journal of Advanced Manufacturing Technology* 37(9):872-880, 2008. doi:https://doi.org/10.1007/s00170-007-1034-8.
8. Enrico, D.L., Tinne, D.L., and Herman, B., "Hierarchical Dirichlet Process Hidden Markov Models for Abnormality Detection in Robotic Assembly," *Proceedings of Neural Information Processing Systems,* Dec. 3-8, 2012, Lake Tahoe, NV, 2012.
9. Rojas, J., Harada, K., Onda, H., Yamanobe, N. et al., "A Relative-Change-Based Hierarchical Taxonomy for Cantilever-Snap Assembly Verification," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 7-12, 2012, Vilamoura, Portugal, 356-363. IEEE, 2012.
10. Stolt, A., Linderoth, M., Robertsson, A., and Johansson, R., "Detection of Contact Force Transients in Robotic Assembly," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 26-30, 2015, Seattle, WA, 962-968. IEEE, 2015.
11. ABB Homepage, "Integrated Force Control," https://library.e.abb.com/public/0bd0c7f682b2c035c1257cd90028b4e4/Integrated%20Force%20Control_External.pdf, accessed July 15, 2017.
12. GITHUB homepage, "LightGBM Features," https://github.com/Microsoft/LightGBM/wiki/Features, accessed Aug. 1, 2017.
13. Shi, H., "Best-First Decision Tree Learning," Dissertation, The University of Waikato, Hamilton, 2007.

SAE INTERNATIONAL

ARTICLE INFO
Article ID: 05-11-01-0001.1
doi:10.4271/05-11-01-0001.1


# ERRATUM

**Reference of the article:** Zhao, R., Drouot, A., and Ratchev, S., "Classification of Contact Forces in Human-Robot Collaborative Manufacturing Environments," *SAE Int. J. Mater. Manf.* 11(1):2018.

**Erratum Date: 17 May 2018**

**Correction requested: by Author**

**Date of request: 26 April 2018**

**History:**

1. The author affiliation for Adrien Drouot should be the University of Franche-Comté.